

IBM System 36/38 Work Station Emulation Program Version 1.0 Technical Reference

Office Systems Family



**IBM System 36/38
Work Station Emulation
Program Version 1.0
Technical Reference**

Office Systems Family



First Edition (June 1987)

This publication could include technical inaccuracies or typographical errors. Changes are periodically made to the information herein; these changes will be incorporated in new editions of the publication. IBM may make improvements and/or changes in the product(s) and/or programs(s) described in this publication at any time.

The numbers at the bottom right of illustrations are publishing control numbers and are not part of the technical content of this manual.

References in this publication to IBM products, programs, or services do not imply that IBM intends to make these available in all countries in which IBM operates. Any reference to an IBM licensed program in this publication is not intended to state or imply that only IBM's licensed program may be used. Any functionally equivalent program may be used instead.

Products are not stocked at the address below. Requests for copies of this publication and for technical information about IBM Personal System/2 products should be made to your authorized IBM Personal System/2 dealer or your IBM Marketing Representative.

The following paragraph applies only to the United States and Puerto Rico:

A Product Comment Form is provided at the back of this publication. If the form has been removed, address comments to: IBM Corporation, Information Development, Department 245, Rochester, Minnesota, U.S.A. 55901. IBM may use or distribute any of the information you supply in any way it believes appropriate without incurring any obligations whatever.

Contents

Preface	v
Organization of This Manual	vi
Related Manuals	vii
Data Security	viii
Technical Assistance	ix
 Chapter 1. Technical Considerations	1-1
Introduction	1-2
Interface with the Host System	1-2
Interface with Application Programs	1-3
BASIC Application Program Interface Access	1-4
Fixed Address Vector to Key Emulator Functions	1-4
 Chapter 2. Attribute Appearances	2-1
Introduction	2-2
Monochrome Displays	2-2
Color Displays	2-6
 Chapter 3. Using Printers	3-1
Introduction	3-2
Character Sets	3-2
Forms Handling	3-3
EBCDIC-to-ASCII Translation	3-4
5219 Printer Commands	3-5
5224 Printer Commands	3-12
Commands Not Supported	3-12
Commands Supported	3-12
Error Conditions	3-15
 Chapter 4. Using the Printer Function Table Setup Program	4-1
Introduction	4-2
PFTs Provided by the Emulation Program	4-2
Creating a Printer Function Table	4-3
Entering Control Sequences	4-4
Variable N	4-5
Defining Printer Functions	4-6
Initialization and Reset	4-7
Vertical and Horizontal Spacing	4-8

Highlighting	4-11
Paper and Page Options	4-11
Type Styles and Character Sets	4-13
Function Selection Tests	4-15
Chapter 5. Graphics Data Stream	5-1
Introduction	5-2
Fully Supported and No-op Orders	5-4
Modified Orders	5-4
Graphics Control Orders	5-5
Graphics Set Orders	5-8
Graphics Draw Orders	5-21
Read Orders	5-23
Printer Orders	5-28
IEEE Interface Orders	5-29
Chapter 6. Graphics Display	6-1
Introduction	6-2
Graphics Display Differences	6-3
Device Driver Specifications	6-6
Appendix A. Character Sets	A-1
IBM Personal Computer Printer Character Conversion	A-2
Scan Codes	A-3
IBM Personal System/2 Keyboards and Scan Codes	A-3
IBM 5250 Keyboard and Scan Codes	A-7
Character Set Tables	A-9
EBCDIC Character Set Tables	A-11
ASCII Character Sets	A-21
EBCDIC-to-ASCII Conversion Tables	A-25
Appendix B. Decimal to Hexadecimal Conversion	B-1
Appendix C. Updating the Emulation Program	C-1
Introduction	C-2
The Control Data Set	C-2
Updating with the Update Program	C-5
Listing Applied Updates	C-6
Glossary	G-1
Index	X-1

Preface

This manual provides information about the technical aspects and advanced functions that are available in the IBM System 36/38 Work Station Emulation Program for an IBM PERSONAL SYSTEM/2™ for use on a host system that supports the 5250 Information Display System.

This manual uses the following conventions:

- IBM Personal System/2 refers to any or all of the family of IBM Personal System/2s: Models 50, 60, and 80.
- Host refers to the IBM System/36 or the IBM System/38.
- The IBM Work Station Emulation Program is referred to as the Emulation program.
- The IBM System 36/38 Work Station Emulation Adapter is referred to as the adapter.
- The *IBM System 36/38 Work Station Emulation Program User's Guide* is referred to as the *User's Guide*.
- Information about the IBM 5224 Printer is also valid for the IBM 5225 Printer.
- A function or character that is not available on the IBM Personal System/2 is described as an emulated function or emulated character, respectively.

The Emulation program enables you to imitate the functions of the IBM 3196 Display Station and IBM 5219, 5224, 5225, and 5256 Printers and to use the resources and functions of your host.

Graphics functions are supported in a way similar to the IBM 5292 Model 2 Color Display Station. A plotter is also supported. The graphics functions can be used in any display session.

The adapter and Emulation program allow the IBM Personal System/2 to perform the function of multiple work stations at the same time on one twinaxial cable.

Organization of This Manual

This manual contains six chapters and three appendixes.

- Chapter 1, "Technical Considerations," supplies technical information for the Emulation program.
- Chapter 2, "Attribute Appearances," compares the screen attributes for the IBM Personal System/2 display with those for the IBM 3196 display.
- Chapter 3, "Using Printers," supplies technical information for emulation of the IBM 5219 Printer (Models D01/D02), IBM 5224 Printer, and the IBM 5256 Matrix Printer.
- Chapter 4, "Using the Printer Function Table Setup Program," describes how to create the printer function tables necessary for user-defined printers.
- Chapter 5, "Graphics Data Stream," describes the graphics orders that are different from those described in the *IBM 5250 Information Display System Functions Reference Manual*.
- Chapter 6, "Graphics Display," presents display differences between the Emulation program and the 5292 Model 2 Color Display.
- Appendix A, "Character Sets," lists the hexadecimal values for the character codes in the IBM Personal System/2 ASCII and host EBCDIC data streams.
- Appendix B, "Decimal to Hexadecimal Conversion" contains tables for converting decimal values to hexadecimal.
- Appendix C, "Updating the Emulation Program" explains how to update the Emulation program.
- Glossary.
- Index.

Related Manuals

Before using the Emulation program, you should have some knowledge of your IBM Personal System/2 (including IBM Personal Computer Disk Operating System), the IBM 5250 Information Display System, and your host system. This manual will not teach you how to use the host functions. Therefore, it is important that you read and understand the following manuals.

The following manuals accompany the Emulation program package:

- *IBM System 36/38 Work Station Emulation Program User's Guide, SC21-8072*
- *IBM System 36/38 Work Station Emulation Program Keyboard Templates, SX21-9882*
- *IBM System 36/38 Work Station Emulation Adapter Quick Reference, SA21-9861*

The following manuals are available from your IBM representative or through IBM direct.

- *IBM Personal Computer Disk Operating System*
- *IBM System 36/38 Work Station Emulation Adapter Hardware Maintenance and Service, SY31-9053*
- *IBM System 36/38 Work Station Emulation Adapter Technical Reference, SA21-9862*
- *IBM 3196 Display Station User's Guide, GA18-2482.*
- *IBM 5250 Information Display System Planning and Site Preparation Guide, GA21-9337*
- *IBM 5250 Information Display System Functions Reference Manual, SA21-9247*

The coordinator for your IBM host system should also be able to supply a list of manuals for the host system. Refer to this list for any additional manuals you may require.

Data Security

CAUTION

There is a potential data security risk if you assign identical addresses to multiple devices in the address search list. When using the address search list, the work station addresses can be automatically assigned to sessions on the IBM Personal System/2. If the IBM Personal System/2 or the host system has a failure, the user may be assigned a different work station address during recovery. As a result, the user may be able to access to another user's application after recovery.

To minimize the potential data security risk, you should do the following:

- If you write your own application program(s), the program should require each user to sign on with a security password before the host system will reestablish a session.
- If you use an application program that does not require a security password before reestablishing a session, you should always have the system operator do the following:
 - Cancel the job if either the IBM Personal System/2 or host system failure occurs during the session and the session is lost.
 - Cancel any spool writer job for a printer session, when applicable.

The Emulation program and adapter, in conjunction with an IBM Personal System/2, is a powerful and useful tool to help you with your personal and business information processing needs. As with any information system, inadvertent errors may occur and information may be misused. It is recommended that when processing sensitive or highly valuable information, you take steps to ensure that your data and programs are protected from accidental or unauthorized disclosure, modification, destruction, or misuse. Simple measures, such as removing diskettes when not in use, keeping backup copies of valuable information, or installing the equipment in a secure facility, can go a long way to maintain the integrity and privacy of your information.

Technical Assistance

Technical assistance questions should be directed to your IBM marketing representative or to your support personnel.

Chapter 1. Technical Considerations

Introduction	1-2
Interface with the Host System	1-2
Interface with Application Programs	1-3
BASIC Application Program Interface Access	1-4
Sample BASIC Programs	1-4
Fixed Address Vector to Key Emulator Functions	1-4
General Purpose Subroutine	1-5
Fixed Address Vector Table for Display Sessions	1-11
Fixed Address Vector Table for Printer Sessions	1-22

Introduction

This chapter explains how the Emulation program interfaces with the host and with personal computer application programs.

Interface with the Host System

The IBM Personal System/2 is connected to the host by a twinaxial cable and a connector that permits other IBM Personal System/2s or 5250 compatible work stations to be connected on the same line. Each IBM Personal System/2 can be individually addressed by the host. The most distant work station can be up to 1525 meters (5000 feet) from the host. Up to seven work stations can be operated at the same time on one line. Each IBM Personal System/2 can have multiple sessions in operation at the same time. Each session requires one of the seven work station addresses.

Transmission of data and commands is half duplex and is host initiated.

All keyboard, display, and printer functions are logically controlled by the host programs for any display station or operator function. The Emulation program provides both an operator and a command interface to the attaching host system, allowing explicit control of the keyboard, display, printer, and features.

The command interface allows the user to control the functions of fields on the display screen. Characters, fields, lines, and the entire regeneration buffer can be accessed.

Key scan codes are sent to the host on request. The display station does not translate key codes into display codes. The keyboard characters, scan codes, and codes for displayable characters are shown in Appendix A, "Character Sets."

Interface with Application Programs

You can operate the Emulation program from a personal computer application program running concurrently with the Emulation program. Use the Hot Key function to switch to the DOS session, then load a program that directly accesses the emulation EBCDIC screen buffer or keystroke queue. The emulation characteristics can be modified, such as changing the character set and attribute translation tables.

A personal computer application program allows you to provide any number of unique host characteristics when an IBM Personal System/2 is attached to a host. For example, the sign-on sequence and host application selection methods could be taken over by an application program, relieving an operator of sign-on procedures for the host. However, a password supplied by an application program would become part of that program disk(ette), and would need physical security protection procedures. Lengthy or complex keying sequences can be stored in an application program.

By running a conversion program on the IBM Personal System/2, other IBM Personal System/2 resident information can be converted to a form that can be made available to the host (for example, information which originates from other devices, such as communication adapters).

BASIC Application Program Interface Access

The Emulation program can be accessed from a BASIC language application program as follows:

```
2610 '  
2620 ' This subroutine checks that the Emulation program is loaded.  
2630 ' It also initializes access pointers into the emulator data segment.  
2640 '  
2650 DEG SEG=0                                : 'Point to interrupt vectors  
2652 TSEG = PEEK(&H36)+ 256*PEEK(&H37)         : 'Read IL5 segment address  
2654 IF TSEG = 0 THEN 2694                     : 'If zero, then not running  
2656 DEF SEG = TSEG                           : 'Point to IL5 segment  
2658 Z = PEEK(&H148)+ 256*PEEK(&H149)          : 'Offset of EC level stamp  
2660 X = PEEK(&H140)+ 256*PEEK(&H141)          : 'Offset of EBCDIC to ASCII  
2664 IF CHR$(PEEK(Z+ 1))<>"E" THEN 2694       : 'If not "EC" then not running  
2666 IF CHR$(PEEK(Z+ 2))<>"C" THEN 2694       : 'If not "EC" then not running  
2668 EC$ = "    "                             : 'Read version number  
2670 FOR I=4 TO 7: EC$=EC$+CHR$(PEEK(Z+I)): NEXT I  
2679 Z = PEEK(&H146)+256*PEEK(&H147)          : 'Offset of EBCDIC screen  
2686 PRINT "Emulator version ";EC$;" is running."  
2688 RETURN  
2690 '----- Error exists -----  
2694 BEEP: LOCATE 2,10: PRINT "Emulation program not running.": SYSTEM
```

RB4F001

Sample BASIC Programs

Sample BASIC programs are provided on the Emulation program diskettes. (These programs are identified by the extension BAS.) These programs can be used when you write personal computer application programs to interface with the Emulation program. The sample programs use the above method and can be studied for additional information. See Chapter 2 in the *User's Guide* for a description of these programs.

Commented lines in the source files contain information about the BASIC programs. Source code is provided if you want to modify these programs to meet your requirements. Become familiar with their content and function before you attempt to change or use them.

Fixed Address Vector to Key Emulator Functions

Figure 1-1, Figure 1-3, and Figure 1-4 represent a section of memory in the lowest part of the emulation segment, which points to key resources within the Emulation program. The pointer values are stored with the least significant byte first. Data should not be changed except where the tables indicate that changes are allowed.

In the following tables, bits are numbered 1 through 8, where each bit has the following hex value:

Bit	Hex Value
1	01
2	02
3	04
4	08
5	10
6	20
7	40
8	80

The hexadecimal addresses represents the Emulation program subroutine entry location as an offset into the data segment for each session. The subroutine can be addressed by a personal computer application with a *long* or *far* indirect call to the given address. Each entry in Figure 1-1 indicates whether you can use the fixed address vector for a display session, a printer session, or both. BASIC programs cannot call the subroutine directly.

You should make sure that all registers required by the personal computer application are saved before calling the entry points. All registers are restored upon return unless stated differently.

General Purpose Subroutine

Figure 1-1 lists general purpose functions of the Emulation program. Offset hex 0100 into the first session's segment address contains the 4-byte address of the executable code (see "Fixed Address Vector Table for Display Sessions" on page 1-11). Use a *far* indirect call to access the general purpose subroutine.

Register Contents

AH Equals hex FF (upon return from an inlink) if the function is not supported. It equals hex FE if an input value is out of range.

AH	Name and Description
3	<p>ILSCRNUP: This function updates the ASCII display buffer from the EBCDIC buffer and refreshes the IBM Personal System/2 screen. You can control screen refreshing by checking the SURX flag at address 0164 and then call this subroutine when the flag is nonzero.</p> <p>Input:</p> <p>DS Selected session's segment address (see ILSELSESN subroutine)</p> <p>You can use this function for display and printer sessions; however, printer sessions do not have an EBCDIC buffer.</p>
5	<p>ILSELSESN: This function returns the data segment value and device type for a session identified in register AL. When register AL contains binary zero upon entry to the subroutine, the data segment value for session 1 is returned in register DS and the device type code for session 1 is returned in register AL. The device type code is binary one for a display session or binary two for a printer session.</p> <p>When register AL contains binary one upon entry to the subroutine, values for session 2 are returned. If session 2 was not configured, binary zero is returned for all values.</p> <p>When calling this function, enter a session number (0 to 3) into the AL register.</p> <p>Output:</p> <p>AL = FE out of bounds AL = 0 undefined</p> <p>You can use this function for display and printer sessions.</p>

Figure 1-1 (Part 1 of 5). General Purpose Subroutine

AH	Name and Description
8	<p>ILQRESET: This function will cause the keyboard buffer and link scan code queues to be cleared for a session. Any playback sequence in progress is terminated. The break code for all shift keys and the scan code for the emulated Reset key are sent to the work station controller. This leaves the keyboard in a "hands off" reset state and cancels Insert mode.</p> <p>Input:</p> <p>DS Selected session's segment address (see ILSELSESN subroutine)</p> <p>You can use this function for display and printer sessions.</p>
A	<p>ILPRTCHAR: This function provides one method of buffering ASCII characters or control codes for sending to the attached personal computer printer. Data received this way is given priority over data generated by the data stream conversion process and will be sent to the attached printer first.</p> <p>Input:</p> <p>CX String length (maximum of 321 bytes)</p> <p>DS Selected session's segment address (see ILSELSESN subroutine)</p> <p>ES:DI Address of string</p> <p>You can use this function for printer sessions only.</p>

Figure 1-1 (Part 2 of 5). General Purpose Subroutine

AH	Name and Description
B	<p>ILPRTOPTN: This function lets you process an emulated printer option. This function is identical to selecting the corresponding option from the emulated printer operator panel.</p> <p>Input:</p> <p>AL = 1 Start AL = 2 Stop AL = 3 Suspend AL = 4 Cancel</p> <p>5256 and 5224 emulation only:</p> <p>AL = 5 Line Feed AL = 6 Form Feed AL = 8 Select 6 LPI AL = 9 Select 8 LPI AL = A Select 10 CPI AL = B Select 15 CPI</p> <p>DS Selected session's segment address (see ILSELSESN subroutine)</p> <p>You can use this function for printer sessions only.</p>

Figure 1-1 (Part 3 of 5). General Purpose Subroutine

AH	Name and Description
C	<p>ILNOSHOW: This function controls how emulation uses the IBM Personal System/2 display.</p> <p>Input:</p> <p>AL = 0 Inhibit screen updates. Although the host sends data to the emulation's EBCDIC buffer, the IBM Personal System/2 ASCII screen is no longer updated to match. This requires the application program to take control of the screen display. No other emulation activity (such as keyboard handling and outlink processing) is affected.</p> <p>AL = 1 Normal screen update processing.</p> <p>DS Selected session's segment address (see ILSELSESN subroutine).</p> <p>You can use this function for display and printer sessions.</p>
E	<p>ILHOTKEY: This function forces a Hot Key operation. This allows the user to control the current session.</p> <p>Input:</p> <p>AL = 0 DOS session</p> <p>AL = 1 Session 1</p> <p>AL = 2 Session 2</p> <p>AL = 3 Session 3</p> <p>AL = 4 Session 4</p> <p>Output:</p> <p>AL = 0 Hot Key successful</p> <p>AL = FF Session not defined or not active</p> <p>You can use this function for display and printer sessions.</p>

Figure 1-1 (Part 4 of 5). General Purpose Subroutine

AH	Name and Description
11	<p>ILSAVE: This function allows an application program to request a screen save. The screen will be saved on the adapter.</p> <p>Input:</p> <p>DS Screen segment CX Byte count</p> <p>Output: All registers are restored.</p>
12	<p>ILRESTOR: This function allows an application program to request a screen restore. The previously saved screen will be restored.</p> <p>Input:</p> <p>ES Screen segment CX Byte count</p> <p>Output: All registers are restored.</p>

Figure 1-1 (Part 5 of 5). General Purpose Subroutine

Figure 1-2 is a sample program that accesses these subroutines.

```

;*****
;* This sample inlink program assumes the emulator is loaded. *
;*****
cseg      segment para public 'code'
          assume  cs:cseg,ds:cseg,es:cseg,ss:cseg
          org     100h
sample:   jmp     start          ; Bypass data area
intseg    dw      0
inlink    dd      0
          db      20 dup('stack')
tos       db      0
;
start     proc     near
          mov     sp,offset tos  ; Set up stack pointer for the program
          mov     ds,intseg      ; Set up es to interrupt vector table
          mov     si,36h         ; Get the emulators first session seg
          mov     ds,[si]        ; Put first session seg in ds
          mov     si,0100h       ; Point to the FAR indirect call addr
          mov     di,offset inlink ; Point to destination for copy
          mov     cx,4           ; Set up length for the copy
rep       movsb                ; Copy the address to my data seg
          mov     ah,5           ; Put inlink function code for finding
                                ; a session data seg and session type
          mov     al,1           ; Set up to inquire about session 2
          call    cs:inlink       ; Do the inlink
;*****
;* On return DS=Session 2 Seg, AL=Device Code *
;*****
          int     20h           ; Exit
;
start     endp
cseg      ends
          end      sample

```

RB4F008

Figure 1-2. Sample Subroutine Access

Fixed Address Vector Table for Display Sessions

The data addresses in Figure 1-3 on page 1-12 are offsets from the session's data segment. The first session data segment value can be found at the segment address 0 (zero), offset hex 36 and 37. If this word is zero, the Emulation program is not running. If this word is not zero, then it is used as a segment paragraph address to read the word at offset hex 0148 of this segment.

Use this word's content as an offset to address yet another field. If the first 3 bytes are 20 45 43 (&bla.EC, where &bla. is a blank), the Emulation program is loaded and the fixed address vector table can be accessed with the segment value found at 0:0036 and 0:0037 hex. The next 5 bytes indicate the release level of code loaded. For example, Release 3 is indicated by hex 30 33 2E 30 30 (03.00). These bytes will change with each new release level of the Emulation program.

Address (Hex)	Name and Description
013E	INDICST: This byte is composed of bits that indicate if any display status indicators have been turned ON since the last time this byte was set to hex 00 by an application program. It is used with the next byte in this table, primarily to find out if the attached host has completed processing a function key that causes the Input Inhibited indicator to be turned ON, then OFF. This byte should be reset to zero before sending a scan code to the host which turns the Input Inhibited indicator ON.
013F	INDICRS: Indicator history bits. Each bit in this byte is set to zero whenever the corresponding indicator has been turned OFF. This byte should be set to hex FF before sending any scan code to the host that causes Input Inhibited to be turned ON and OFF. During normal operations, some emulated host functions may send Shift key scan codes to the host. These scan codes affect the state of the Shift indicator in INDICRS, but they do not affect the Shift indicator displayed on the status line. The displayed status line indicators always reflect the Shift keys pressed by the operator or sent through KEYO. The indicator and bit correspondence is listed below for address 015A.
0140 and 0141	XXLCHR: Pointer to EBCDIC-to-ASCII translation table (256 bytes long). This table is located within the data segment. Use the EBCDIC character code to index this table to access the corresponding ASCII character code. This table is initialized to values provided by the configuration program.

Figure 1-3 (Part 1 of 10). Fixed Address Vectors for Display Sessions

Address (Hex)	Name and Description
0142 and 0143	<p>XXLATT: Pointer to host-to-IBM Personal System/2 attribute translation table (32 bytes long). The host attribute value minus hex 20 is used to index the entries in this table, which are the corresponding IBM Personal System/2 attribute values. Host attribute values range from hex 20 through 3F. Application programs can modify the attribute table to alter the color or appearance of displayed text. This table is initialized according to data provided by the configuration program.</p>
0144 and 0145	<p>XXLKEY: Pointer to the PC-to-5250 scan code translation table (84 bytes long). Each key position has an entry. The first location is hex 02 and indicates the keyboard ID. Application programs can use this table to translate individual scan codes then send them to the host using KEYO. Application programs can modify the table without affecting the Emulation program. The Emulation program uses another table for keystroke translation, which it constructs from the keyboard file when the Emulation program is loaded. This table does not support all PC scan codes on the Enhanced keyboard.</p>
0146 and 0147	<p>XREGEN: Pointer to the start of the EBCDIC screen storage buffer. The starting location corresponds to the upper left screen position and continues character by character for the 1920 visible screen positions. The contents are EBCDIC character codes. Application programs can modify the EBCDIC screen buffer. See the WSESS.BAS and WSEUL.BAS sample programs.</p> <p>Note: The first character of the status line is located at (XREGEN)-80.</p>

Figure 1-3 (Part 2 of 10). Fixed Address Vectors for Display Sessions

Address (Hex)	Name and Description
0148 and 0149	XEC: Pointer to the engineering change level of this program. The change level identifies the version of the program that is currently running. The referenced address is the starting address of an 8-byte field, &bla.ECXX.YY, where EC is the environment, XX is the version and YY is the modification of the program, and &bla. represents a blank.
014A and 014B	Reserved.
014C and 014D	XILVL@: Pointer to the interrupt vector address used by the adapter.
014E	LKTIMSK: Interrupt mask byte for the 8259, which corresponds to the operating interrupt level (hex 20 for interrupt level 5).
014F	NOTMSK: The bit inverted value of LKTIMSK (hex DF for interrupt level 5).
0150	KEYI: The IBM Personal System/2 scan code from each keystroke. Each new code overlays the old code. This byte can be altered by an application program without affecting the Emulation program. See "Scan Codes" on page A-3.
0151	KEYO: Used to send 5250 scan codes to the host. Placing a nonzero value in this byte results in its transmission to the host, as if a key was pressed. This byte is replaced with hex 00 when the code is transmitted to the host. See the WSEASO.BAS sample program, and "Scan Codes" on page A-3.
0152 through 0155	These four bytes can be altered by an application program to find or change the virtual diskette drive names used by the Emulation program. See the WSESVD.BAS sample program.
0152	DSKDI: First drive intercept.
0153	DSKDI0: Second drive intercept.

Figure 1-3 (Part 3 of 10). Fixed Address Vectors for Display Sessions

Address (Hex)	Name and Description															
0154	DSKDI1: Third drive intercept.															
0155	DSKDI2: Fourth drive intercept.															
0156 through 0159	Reserved.															
015A	<p>INDIC: Indicator status byte (indicators are ON if bits are on):</p> <table><tr><th>Bit</th><th>Hex</th><th>Description</th></tr><tr><td>2</td><td>02</td><td>Input Inhibited</td></tr><tr><td>4</td><td>08</td><td>Insert mode</td></tr><tr><td>6</td><td>20</td><td>Shift mode</td></tr><tr><td>8</td><td>80</td><td>Message Waiting</td></tr></table> <p>All other bits are reserved.</p>	Bit	Hex	Description	2	02	Input Inhibited	4	08	Insert mode	6	20	Shift mode	8	80	Message Waiting
Bit	Hex	Description														
2	02	Input Inhibited														
4	08	Insert mode														
6	20	Shift mode														
8	80	Message Waiting														
015B	SYSAB: System Available indicator status. If not zero, the System Available indicator is ON.															

Figure 1-3 (Part 4 of 10). Fixed Address Vectors for Display Sessions

Address (Hex)	Name and Description		
015C	BYP5: Automatic sign-on/keyboard control override.		
	<i>Bit</i>	<i>Hex</i>	<i>Description</i>
	1	01	Reserved.
	2	02	Bypass this session when selected by the Hot Key.
	3	04	Reserved.
	4	08	If this bit is on, keystroke translation only. Codes are not sent to the host. Translated scan codes are stored in LASTKEY.
	5	10	If bit is not zero, sends all translated keys to host. Keystrokes in the DOS session are sent to the host, regardless of the state of Hot Key, bit 6, or BIOS processing.
	6	20	Send all keys to BIOS. If not zero, all keys are processed normally by the Emulation program regardless of Hot Key or bit 5. If the normal keyboard interrupt vector pointed to BIOS when the Emulation program was loaded, BIOS keystroke buffering is limited and is not emptied until the user switches to the DOS session. Keys can be passed to both the application program and the host system by turning bits 5 and 6 on together. This allows an application program to monitor keys being sent to the host. Keystroke recording and soft key definition are easier to implement with these functions. It is recommended that bit 7 also be turned on when using this function.

Figure 1-3 (Part 5 of 10). Fixed Address Vectors for Display Sessions

Address (Hex)	Name and Description	
015C continued	7 40	Hot Key to emulation mode is inhibited for this session. If this is not zero, this session is removed from the Hot Key function. The session continues to communicate with the host. This function allows an application program to control an emulation session without interference from the user during a Hot Key function. DOS may not be prohibited in this manner.
	8 80	Reserved.
015D and 015E	CURSOR: Binary row and column cursor position. Column value is in 015D, row is in 015E. Column ranges from 0 to 79, row from 0 to 23. Changing the value in these addresses does not change the screen position of the cursor.	
015F and 0160	Reserved.	
0161 and 0162	CRT_SAV: Pointer to application screen save and restore buffer. The value is an offset from the beginning of the Emulation program. The save/restore buffer is 4K when a monochrome monitor is installed and 12K when a color monitor is installed. You can configure to save the screen on the adapter and reduce the Emulation program by 12K. This buffer is overwritten by the Emulation program when the user switches from DOS to an emulation session.	

Figure 1-3 (Part 6 of 10). Fixed Address Vectors for Display Sessions

Address (Hex)	Name and Description						
0163	PRTBC: Printer character set boundary check value. Characters whose ASCII code is above this value are printed as blanks. The replacement function occurs for characters processed for an IBM Personal System/2 print screen request during a display session. This byte can be modified by application programs to inhibit printing special characters. This byte is initialized according to configuration of the print screen character set.						
0164	SURX: Externally available screen change flag. This byte is nonzero when this session's EBCDIC screen buffer changes. The byte can be altered by application programs without affecting the Emulation program.						
0165	Reserved.						
0166	VRDO: Virtual versus real diskette intercept request override.						
0167	DVC_TYP: Device model ID byte. This byte is hex C2 for display emulation.						
0168	<p>MODEL_ID: Display type ID byte.</p> <table> <tr> <td><i>ID</i></td><td><i>Hex Value</i></td></tr> <tr> <td>5292-2</td><td>49</td></tr> <tr> <td>3196</td><td>18</td></tr> </table>	<i>ID</i>	<i>Hex Value</i>	5292-2	49	3196	18
<i>ID</i>	<i>Hex Value</i>						
5292-2	49						
3196	18						
0169 through 016C	Reserved.						
016D	LASTKEY: The last 5250 scan code sent to the host. Not to be used with KEYO operations. This byte can be modified by application programs without affecting emulation. When IBM Personal System/2 keystrokes are translated to a string of more than one 5250 scan code, they appear in this byte in rapid succession. See "Scan Codes" on page A-3.						

Figure 1-3 (Part 7 of 10). Fixed Address Vectors for Display Sessions

Address (Hex)	Name and Description
016E through 0175	Reserved.
0176 and 0177	<p>PCREGN@: Segment pointer to IBM Personal System/2 monochrome or color video display buffer. These bytes are in segment (paragraph) form, to be loaded into a segment register. These bytes control the address at which the Emulation program normally directs video output. For example, the monochrome display supports a display buffer at hex B000.</p> <p>These bytes are normally zero to allow the Emulation program to use the current display buffer as determined by the BIOS equipment byte. An application program can make these bytes nonzero, to override the current display buffer.</p>

Figure 1-3 (Part 8 of 10). Fixed Address Vectors for Display Sessions

Address (Hex)	Name and Description															
0178	<p>FLAGS: Hot Key control flags. Reserved bit positions should not be modified by an application program.</p> <table><tr><th>Bit</th><th>Hex</th><th>Description</th></tr><tr><td>1</td><td>01</td><td>Hot Key from DOS session to emulation mode. Set on by an application program to request the Hot Key. Turn this bit on, and execute an INT 09 to switch to emulation mode.</td></tr><tr><td>2</td><td>02</td><td>Programmed Hot Key. Made nonzero by an application program to cause the Emulation program to return to the DOS session after the System Available indicator is turned ON.</td></tr><tr><td>3</td><td>04</td><td>Reserved.</td></tr><tr><td>5</td><td>10</td><td>Reserved.</td></tr></table>	Bit	Hex	Description	1	01	Hot Key from DOS session to emulation mode. Set on by an application program to request the Hot Key. Turn this bit on, and execute an INT 09 to switch to emulation mode.	2	02	Programmed Hot Key. Made nonzero by an application program to cause the Emulation program to return to the DOS session after the System Available indicator is turned ON.	3	04	Reserved.	5	10	Reserved.
Bit	Hex	Description														
1	01	Hot Key from DOS session to emulation mode. Set on by an application program to request the Hot Key. Turn this bit on, and execute an INT 09 to switch to emulation mode.														
2	02	Programmed Hot Key. Made nonzero by an application program to cause the Emulation program to return to the DOS session after the System Available indicator is turned ON.														
3	04	Reserved.														
5	10	Reserved.														
0179 and 017A	Reserved.															
017B	HKCOL: Hot Key column (1 through 80). This byte and the next two work together to allow host programs to cause an automatic Hot Key to the DOS session. The Emulation program automatically switches to the DOS session when the specified EBCDIC character (HKCHAR) is found in the selected row and column (HKROW, HKCOL). The function is disabled when HKCOL is zero.															
017C	HKROW: Hot Key row (1 through 24).															
017D	HKCHAR: Hot Key character in EBCDIC.															
017E through 0181	Reserved.															

Figure 1-3 (Part 9 of 10). Fixed Address Vectors for Display Sessions

Address (Hex)	Name and Description
0182 and 0183	CODESEG: Segment address of Emulation program code. The value is in segment (paragraph) form, for use in segment registers. This word identifies the location where the code is loaded. The code segment is assigned by DOS and depends on the number and types of IBM Personal System/2 resources in use when the program is loaded. There is no information useful to other application programs in this segment.
0184	SESSNO: Session number. Each session's number is uniquely identified in this byte by the Emulation program.
0185 through 0192	Reserved.
0193	STN@: Station address. The address (one of hex 0 through 6) this session uses to respond to the host.
0198 through 0199	NSDS: Next session's data area. Segment pointer to the data area for the next configured session. Pointer is in segment (paragraph) form, for use in segment registers. The last session's entry points to the first session's data area, forming a circular linked list. Application programs can use this pointer to locate the data for each session defined for the Emulation program.

Figure 1-3 (Part 10 of 10). Fixed Address Vectors for Display Sessions

Fixed Address Vector Table for Printer Sessions

The display and printer data areas have similar formats, but they are not identical. The printer session only uses the following data areas. All data areas not listed in Figure 1-4 are reserved in a printer session. Unless specifically mentioned otherwise, storage locations must not be modified by application programs.

Address (Hex)	Name and Description
0140 and 0141	XXLCHR: Pointer to EBCDIC-to-ASCII translation table (192 bytes long). This table is located within the emulator's data segment, and begins with EBCDIC code point hex 40. Use the EBCDIC character code to index this table to access the corresponding ASCII character code. The table is initialized to values provided by the configuration program.
0142 and 0143	XXLATT: Pointer to host-to-IBM Personal System/2 attribute translation table (32 bytes long). The host attribute value minus hex 20 is an index into this table, which contains the corresponding IBM Personal System/2 attribute values. Host attribute values range from hex 20 to hex 3F. You can modify the attribute table to alter the color or appearance of the emulated operator panel. This table is initialized to values provided by the configuration program.
0148 and 0149	XEC: Pointer to the engineering change level of this program. This gives the version of the program that is currently running. The referenced address is the starting address of a 8-byte ASCII field, &bla.ECXX.YY, where EC is the environment, XX is the version and YY is the modification of the program, and &bla. represents a blank.

Figure 1-4 (Part 1 of 8). Fixed Address Vectors for Printer Sessions

Address (Hex)	Name and Description																											
015C	<p>BYPS: Hot Key controls.</p> <table><tr><th>Bit</th><th>Hex</th><th>Description</th></tr><tr><td>1</td><td>01</td><td>Reserved.</td></tr><tr><td>2</td><td>02</td><td>Reserved.</td></tr><tr><td>3</td><td>04</td><td>Reserved.</td></tr><tr><td>4</td><td>08</td><td>Reserved.</td></tr><tr><td>5</td><td>10</td><td>Reserved.</td></tr><tr><td>6</td><td>20</td><td>Reserved.</td></tr><tr><td>7</td><td>40</td><td>Hot Key inhibit for this session. If this bit is nonzero, this session is removed from the Hot Key function. The session continues to communicate with the host system. This allows an application program to control an emulation session without incidental interference from the user during a Hot Key function. There is no provision to inhibit the DOS session in this fashion.</td></tr><tr><td>8</td><td>80</td><td>Reserved.</td></tr></table>	Bit	Hex	Description	1	01	Reserved.	2	02	Reserved.	3	04	Reserved.	4	08	Reserved.	5	10	Reserved.	6	20	Reserved.	7	40	Hot Key inhibit for this session. If this bit is nonzero, this session is removed from the Hot Key function. The session continues to communicate with the host system. This allows an application program to control an emulation session without incidental interference from the user during a Hot Key function. There is no provision to inhibit the DOS session in this fashion.	8	80	Reserved.
Bit	Hex	Description																										
1	01	Reserved.																										
2	02	Reserved.																										
3	04	Reserved.																										
4	08	Reserved.																										
5	10	Reserved.																										
6	20	Reserved.																										
7	40	Hot Key inhibit for this session. If this bit is nonzero, this session is removed from the Hot Key function. The session continues to communicate with the host system. This allows an application program to control an emulation session without incidental interference from the user during a Hot Key function. There is no provision to inhibit the DOS session in this fashion.																										
8	80	Reserved.																										
0161 and 0162	<p>INITSTRNG@: Pointer to 64-byte printer initialization string. The value represents an offset from this session's data segment to the printer initialization string. This string contains 64 bytes of ASCII characters and control codes that will be output to the attached printer whenever printer emulation is started after being suspended. This string can be configured by the user and modified by an application program using this pointer. This pointer, however, must not be modified.</p>																											
0164	<p>SURX: Externally available screen change flag. Printer emulation sets this byte nonzero when this session's emulated operator panel requires refreshing. Application programs can alter this byte without affecting printer emulation.</p>																											

Figure 1-4 (Part 2 of 8). Fixed Address Vectors for Printer Sessions

Address (Hex)	Name and Description																		
0166	<p>ATTDEVID: Attached printer device type. This byte value identifies the type of personal computer printer the user specified as being attached for this printer session.</p> <table><tr><th>Bit</th><th>Hex</th><th>Description</th></tr><tr><td>2</td><td>02</td><td>IBM Graphics Printer</td></tr><tr><td>4</td><td>08</td><td>IBM Proprinter</td></tr><tr><td>5</td><td>10</td><td>IBM Wheelprinter</td></tr><tr><td>7</td><td>40</td><td>IBM Quietwriter</td></tr><tr><td>8</td><td>80</td><td>User-defined printer</td></tr></table>	Bit	Hex	Description	2	02	IBM Graphics Printer	4	08	IBM Proprinter	5	10	IBM Wheelprinter	7	40	IBM Quietwriter	8	80	User-defined printer
Bit	Hex	Description																	
2	02	IBM Graphics Printer																	
4	08	IBM Proprinter																	
5	10	IBM Wheelprinter																	
7	40	IBM Quietwriter																	
8	80	User-defined printer																	
0167	<p>DVC_TYP: Arctic device type identification byte. This byte is hex 2D for 5219 Printer emulation, hex 24 for 5224 Printer emulation, and hex 60 for 5256 Printer emulation.</p>																		
0168	<p>MODEL_ID: Arctic device model identification byte. Use this byte with DVC_TYP to further identify the type of printer emulation.</p>																		
0182 and 0183	<p>CODESEG: Segment address of emulation code. This is in segment (paragraph) form, for use in segment registers. This word identifies the load point of the resident portion of the Emulation program. The code segment is assigned by DOS, and its value depends on the number and types of IBM Personal System/2 resources in use when the Emulation program is loaded. There is no information useful to application programs in this segment.</p>																		
0184	<p>SESSNO: Session number. Each session's number is uniquely identified in this byte.</p>																		
0193	<p>STN@: Station address. This byte contains the twinaxial address (hex 00 through hex 06) at which this session will respond to the host.</p>																		

Figure 1-4 (Part 3 of 8). Fixed Address Vectors for Printer Sessions

Address (Hex)	Name and Description									
0198 and 0199	NSDS: Next session's data area. This is a segment pointer to the data area for the next session. This is in segment (paragraph) form, for use in segment registers. The last session's entry points to the first session's data area, forming a circular linked list. Application programs can use this to locate easily the data for each of the sessions defined to the Emulation program.									
01C5 and 01C6	RCBUF1@: Pointer to Receive Buffer 1. This value represents an offset from this session's data segment to the start of the first 256-byte receive buffer.									
01C7 and 01C8	RCBUF2@: Pointer to Receive Buffer 2. This value represents an offset from this session's data segment to the start of the second 256-byte receive buffer.									
01C9 and 01CA	<p>BUFxFULL@: Pointer to Receive Buffer Flag Byte. This value represents an offset from this session's data segment to the start of the Receive Buffer Flag byte which is defined as follows:</p> <table><tr><th>Bit</th><th>Hex</th><th>Description</th></tr><tr><td>1</td><td>01</td><td>Receive buffer 1 contains data to be processed.</td></tr><tr><td>2</td><td>02</td><td>Receive buffer 2 contains data to be processed.</td></tr></table> <p>Bits 3 through 8 are reserved.</p>	Bit	Hex	Description	1	01	Receive buffer 1 contains data to be processed.	2	02	Receive buffer 2 contains data to be processed.
Bit	Hex	Description								
1	01	Receive buffer 1 contains data to be processed.								
2	02	Receive buffer 2 contains data to be processed.								
01CB and 01CC	RCBUF1CT@: Pointer to Receive Buffer 1 byte count. This value represents an offset from this session's data segment to the start of the Receive Buffer 1 byte count. This byte count is loaded each time new data is received in Receive Buffer 1.									

Figure 1-4 (Part 4 of 8). Fixed Address Vectors for Printer Sessions

Address (Hex)	Name and Description																		
01CD and 01CE	RCBUF2CT@: Pointer to Receive Buffer 2 byte count. This value represents an offset from this session's data segment to the start of the Receive Buffer 2 byte count. This byte count is loaded each time new data is received in Receive Buffer 2.																		
01D1	<p>PTRDEVNAM: Printer device name. This byte identifies the physical device port being used by printer emulation. The high-order nibble identifies the port type (serial or parallel), while the low-order nibble identifies the specific port of that type as defined below:</p> <table> <tr> <th>Hex</th><th>Port</th></tr> <tr> <td>00</td><td>LPT1</td></tr> <tr> <td>01</td><td>LPT2</td></tr> <tr> <td>02</td><td>LPT3</td></tr> <tr> <td>03</td><td>LPT4</td></tr> <tr> <td>10</td><td>COM1</td></tr> <tr> <td>11</td><td>COM2</td></tr> <tr> <td>12</td><td>COM3</td></tr> <tr> <td>13</td><td>COM4</td></tr> </table>	Hex	Port	00	LPT1	01	LPT2	02	LPT3	03	LPT4	10	COM1	11	COM2	12	COM3	13	COM4
Hex	Port																		
00	LPT1																		
01	LPT2																		
02	LPT3																		
03	LPT4																		
10	COM1																		
11	COM2																		
12	COM3																		
13	COM4																		

Figure 1-4 (Part 5 of 8). Fixed Address Vectors for Printer Sessions

Address (Hex)	Name and Description		
01D2	SERPARM: Serial parameter byte. This byte identifies the communications parameters being used for communication with a serial printer as defined below. If no serial printer is being used, this value will be zero.		
	<i>Bit(s)</i>	<i>Contents</i>	<i>Value</i>
	8, 7, 6	000	110 baud rate
		001	150 baud rate
		010	300 baud rate
		011	600 baud rate
		100	1200 baud rate
		101	2400 baud rate
		110	4800 baud rate
		111	9600 baud rate
	5, 4	X0	No parity
		01	Odd parity
		11	Even parity
	3	0	1 stop bit
		1	2 stop bits
	2, 1	10	7-bit word length
		11	8-bit word length

Figure 1-4 (Part 6 of 8). Fixed Address Vectors for Printer Sessions

Address (Hex)	Name and Description																																																						
01D4 and 01D5	<p>INDICAT@: Pointer to Indicator State flags. This value represents an offset from the start of this session's data segment to the Indicator State flags. The bits within this byte represent the state of each of the indicators on the emulated operator panel. Indicators are on if the corresponding bits are on. This byte should not be modified.</p> <p>The bits are defined as follows for 5256 Printer emulation.</p> <table><tr><th>Bit</th><th>Hex</th><th>Description</th></tr><tr><td>1</td><td>01</td><td>Reserved</td></tr><tr><td>2</td><td>02</td><td>Reserved</td></tr><tr><td>3</td><td>04</td><td>System Available</td></tr><tr><td>4</td><td>08</td><td>Data Cleared</td></tr><tr><td>5</td><td>10</td><td>Graphics Check</td></tr><tr><td>6</td><td>20</td><td>Forms</td></tr><tr><td>7</td><td>40</td><td>Attention</td></tr><tr><td>8</td><td>80</td><td>Ready</td></tr></table> <p>The bits are defined as follows for 5224 Printer emulation.</p> <table><tr><th>Bit</th><th>Hex</th><th>Description</th></tr><tr><td>1</td><td>01</td><td>Reserved</td></tr><tr><td>2</td><td>02</td><td>Reserved</td></tr><tr><td>3</td><td>04</td><td>System Available</td></tr><tr><td>4</td><td>08</td><td>Change Font</td></tr><tr><td>5</td><td>10</td><td>Graphics Check</td></tr><tr><td>6</td><td>20</td><td>Forms</td></tr><tr><td>7</td><td>40</td><td>Attention</td></tr><tr><td>8</td><td>80</td><td>Ready</td></tr></table>	Bit	Hex	Description	1	01	Reserved	2	02	Reserved	3	04	System Available	4	08	Data Cleared	5	10	Graphics Check	6	20	Forms	7	40	Attention	8	80	Ready	Bit	Hex	Description	1	01	Reserved	2	02	Reserved	3	04	System Available	4	08	Change Font	5	10	Graphics Check	6	20	Forms	7	40	Attention	8	80	Ready
Bit	Hex	Description																																																					
1	01	Reserved																																																					
2	02	Reserved																																																					
3	04	System Available																																																					
4	08	Data Cleared																																																					
5	10	Graphics Check																																																					
6	20	Forms																																																					
7	40	Attention																																																					
8	80	Ready																																																					
Bit	Hex	Description																																																					
1	01	Reserved																																																					
2	02	Reserved																																																					
3	04	System Available																																																					
4	08	Change Font																																																					
5	10	Graphics Check																																																					
6	20	Forms																																																					
7	40	Attention																																																					
8	80	Ready																																																					

Figure 1-4 (Part 7 of 8). Fixed Address Vectors for Printer Sessions

Address (Hex)	Name and Description																											
01D4 and 01D5 continued	<p>The bits are defined as follows for 5219 Printer emulation.</p> <table><tr><th>Bit</th><th>Hex</th><th>Description</th></tr><tr><td>1</td><td>01</td><td>Reserved</td></tr><tr><td>2</td><td>02</td><td>Reserved</td></tr><tr><td>3</td><td>04</td><td>System Available</td></tr><tr><td>4</td><td>08</td><td>Change Font</td></tr><tr><td>5</td><td>10</td><td>Change Setup</td></tr><tr><td>6</td><td>20</td><td>Forms</td></tr><tr><td>7</td><td>40</td><td>Printer Exception</td></tr><tr><td>8</td><td>80</td><td>Ready</td></tr></table>	Bit	Hex	Description	1	01	Reserved	2	02	Reserved	3	04	System Available	4	08	Change Font	5	10	Change Setup	6	20	Forms	7	40	Printer Exception	8	80	Ready
Bit	Hex	Description																										
1	01	Reserved																										
2	02	Reserved																										
3	04	System Available																										
4	08	Change Font																										
5	10	Change Setup																										
6	20	Forms																										
7	40	Printer Exception																										
8	80	Ready																										
01D6 and 01D7	<p>MESSAG@: Pointer to Message State flags. This value represents an offset from the start of the session's data segment to the Message State flags. The bits within this byte represent the state of each of the user messages on the printer operator panel. Messages are being displayed if the corresponding bits are on. This byte should not be modified. The bits are defined as follows.</p> <table><tr><th>Bit</th><th>Hex</th><th>Description</th></tr><tr><td>1</td><td>01</td><td>Invalid Option</td></tr><tr><td>2</td><td>02</td><td>Attached Printer Offline</td></tr><tr><td>3</td><td>04</td><td>Suspended</td></tr><tr><td>4</td><td>08</td><td>Reserved</td></tr><tr><td>5</td><td>10</td><td>Reserved</td></tr><tr><td>6</td><td>20</td><td>Reserved</td></tr><tr><td>7</td><td>40</td><td>Reserved</td></tr><tr><td>8</td><td>80</td><td>Reserved</td></tr></table>	Bit	Hex	Description	1	01	Invalid Option	2	02	Attached Printer Offline	3	04	Suspended	4	08	Reserved	5	10	Reserved	6	20	Reserved	7	40	Reserved	8	80	Reserved
Bit	Hex	Description																										
1	01	Invalid Option																										
2	02	Attached Printer Offline																										
3	04	Suspended																										
4	08	Reserved																										
5	10	Reserved																										
6	20	Reserved																										
7	40	Reserved																										
8	80	Reserved																										

Figure 1-4 (Part 8 of 8). Fixed Address Vectors for Printer Sessions

Chapter 2. Attribute Appearances

Introduction	2-2
Monochrome Displays	2-2
Color Displays	2-6

Introduction

The following figures show the default display attributes for the IBM Personal System/2 monochrome and color displays, compared with the IBM 3196 displays.

You can change the attributes with the configuration program. See the *User's Guide* for instructions.

Monochrome Displays

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
20	Normal	Green characters on black field	White characters on black field
21	Reverse	Black characters on green field	Black characters on white field
22	High intensity	Brighter characters on black field	Brighter characters on black field
23	High intensity and reverse	Black characters on brighter field	Black characters on white field
24	Underscore	Underscore	Underscore
25	Underscore and reverse	Black characters on green field, black underscore	White characters on black field, green underscore
26	Underscore and high intensity	Bright characters on black field, bright underscore	Bright characters on black field, bright underscore
27	Blank screen	Solid black field, no characters	Solid black field, no characters

Figure 2-1 (Part 1 of 5). Monochrome Display Attributes

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
28	Blink	Blinking characters on black field	Blinking characters on black field
29	Blink and reverse	Blinking black characters on green field	Blinking black characters on white field
2A	Blink and high intensity	Blinking bright characters on black field	Blinking bright characters on black field
2B	Blink, high intensity, and reverse	Blinking black characters on bright field	Blinking black characters on white field
2C	Blink and underscore	Blinking characters on black field, green underscore	Blinking characters on black field, white underscore
2D	Blink, underscore, and reverse	Blinking black characters on green field, black underscore	Blinking green characters on black field, white underscore
2E	Blink, underscore, and high intensity	Blinking bright characters on black field, bright underscore	Blinking bright characters on black field, bright underscore
2F	Blank screen	Solid black field, no characters	Solid black field, no characters
30	Column separators	Vertical bar between green characters on black field	White rectangles in null or blank positions on black field
31	Column separators and reverse	Vertical black bars between black characters on green field	Black rectangles with black characters on white field

Figure 2-1 (Part 2 of 5). Monochrome Display Attributes

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
32	Column separators and high intensity	Bright bars between bright characters on black field	Bright rectangle for nulls or blanks with bright characters on black field
33	Column separators, high intensity, and reverse	Black bars between black characters on bright field	Black rectangle for nulls or blanks with black characters on light field
34	Column separators and underscore	Green bars between green underscored characters on black field	White rectangle for nulls or blanks with white underscored characters on black field
35	Column separators, underscore, and reverse	Black bars between black characters on green field, black underscore	White rectangle for nulls or blanks with white characters on black field, black underscore
36	Column separators, underscore, and high intensity	Bright bars between bright characters on black field	Bright rectangle for nulls or blanks with bright characters on black field, black underscore
37	Column separators and no characters	Green bars on black field, no characters	White rectangles for nulls or blanks in field, no characters
38	Column separators and blink	Green bars between blinking characters on black field	Blinking rectangles for nulls or blanks with blinking characters on black field

Figure 2-1 (Part 3 of 5). Monochrome Display Attributes

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
39	Column separators, blink, and reverse	Black bars between blinking black characters on a green field	Blinking black rectangle for nulls or blanks with blinking black characters on white field
3A	Column separators, blink, and high intensity	Bright bars between blinking bright characters on black field	Bright rectangle for nulls or blanks with blinking bright characters on black field
3B	Column separators, blink, high intensity, and reverse	Black bars between blinking black characters on a bright field	Blinking black rectangle for nulls or blanks with blinking white field
3C	Column separators, blink, and underscore	Green bars between blinking underscored characters on black field	Blinking rectangles for nulls or blanks with blinking underscored characters on black field
3D	Column separators, blink, underscore, and reverse	Black bars between blinking black underscored characters on green field	Blinking rectangles for nulls or blanks with blinking characters on black field, underscored
3E	Column separators, blink, underscore, and high intensity	Bright bars between blinking bright underscored characters on black field	Blinking bright rectangles for nulls or blanks with blinking bright characters on black field

Figure 2-1 (Part 4 of 5). Monochrome Display Attributes

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
3F	Column separators and no characters	Green bars on black field, no characters	White rectangles for nulls or blanks on black field, no characters

Figure 2-1 (Part 5 of 5). Monochrome Display Attributes

Color Displays

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
20	Normal	Green characters on black field	Green characters on black field
21	Reverse	Black characters on green field	Black characters on green field
22	High intensity	White characters on black field	White characters on black field
23	High intensity and reverse	Black characters on white field	Black characters on white field
24	Underscore	Green characters on black field with blue underscore	White characters on black field with green underscore characters for nulls
25	Underscore and reverse	Black characters on green field with blue underscore	Black characters on green field with black underscore characters for nulls

Figure 2-2 (Part 1 of 5). Color Display Attributes

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
26	Underscore and high intensity	White characters on black field with blue underscore	White characters on black field with white underscore characters for nulls
27	Blank screen	Solid black, no characters	Solid black, no characters
28	Blink	Red characters on black field	Red characters on black field
29	Blink and reverse	Black characters on red field	Black characters on red field
2A	Blink and high intensity	Blinking red characters on black field	Blinking red characters on black field
2B	Blink, high intensity, and reverse	Blinking black characters on red field	Blinking black characters on red field
2C	Blink and underscore	Red characters on black field with blue underscore	Red characters on black field with red underscore characters for nulls
2D	Blink, underscore, and reverse	Black characters on red field with blue underscore	Black characters on red field with black underscore characters for nulls
2E	Blink, underscore, and high intensity	Blinking red characters on black field with blue underscore	Blinking red characters on black field with blinking red underscore characters for nulls

Figure 2-2 (Part 2 of 5). Color Display Attributes

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
2F	Blank screen	Solid black field, no characters	Solid black field, no characters
30	Column separators	Blue dots in lower corner with turquoise on black characters	Rectangle characters for nulls or blanks on cyan field
31	Column separators and reverse	Blue dots in lower corners with black on turquoise characters	Black rectangle characters with black characters on cyan field
32	Column separators and high intensity	Blue dots in lower corners with yellow characters on black field	Yellow rectangle characters for nulls or blanks with brown characters on black field
33	Column separators, high intensity, and reverse	Blue dots in lower corners with black characters on yellow field	Black rectangle characters for nulls or blanks with black characters on yellow field
34	Column separators and underscore	Segmented blue underscore with turquoise characters on black field	Cyan rectangle characters for nulls or blanks with cyan characters on black field, no underscores
35	Column separators, underscore, and reverse	Segmented blue underscore with black characters on turquoise field	Black rectangle characters for nulls or blanks with black characters on cyan field, no underscore

Figure 2-2 (Part 3 of 5). Color Display Attributes

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
36	Column separators, underscore, and high intensity	Segmented blue underscore with yellow characters on black field	Yellow rectangle characters for nulls or blanks with yellow characters on black field, no underscore
37	Column separators and no characters	No characters and no column separators	Red rectangle characters for nulls or blanks, no other characters
38	Column separators and blink	Pink characters on black field, no column separators	Magenta rectangle characters for nulls or blanks with magenta on black field
39	Column separators, blink, and reverse	Black characters on pink field, no column separators	Black rectangle characters for nulls or blanks with black characters on magenta field
3A	Column separators, blink, and high intensity	Blue characters on black field, no column separators	Blue rectangle characters for nulls or blanks with blue characters on black field
3B	Column separators, blink, high intensity, and reverse	Black characters on blue field, no column separators	Black rectangle characters for nulls or blanks with black characters on blue field

Figure 2-2 (Part 4 of 5). Color Display Attributes

Code	Meaning	IBM 3196 Display	IBM Personal System/2 Display
3C	Column separators, blink, and underscore	Blue underscore with pink characters on black field, no column separators	Magenta rectangle characters for nulls or blanks with magenta characters on black field
3D	Column separators, blink, underscore, and reverse	Blue underscore with black characters on pink field, no column separators	Black rectangle characters for nulls or blanks with black characters on magenta field
3E	Column separators, blink, underscore, and high intensity	Blue underscore with blue characters on black field, no column separators	Blue rectangle characters for nulls or blanks with blue characters on black field
3F	Column separators and no characters	Black field, no characters	Red rectangle characters for nulls or blanks with no other characters

Figure 2-2 (Part 5 of 5). Color Display Attributes

Chapter 3. Using Printers

Introduction	3-2
Character Sets	3-2
Forms Handling	3-3
EBCDIC-to-ASCII Translation	3-4
5219 Printer Commands	3-5
5224 Printer Commands	3-12
Commands Not Supported	3-12
Commands Supported	3-12
Error Conditions	3-15

Introduction

This chapter lists special considerations for using personal computer printers with the Emulation program.

Character Sets

The Emulation program prints the default character defined in the session profile or the default character defined by the Set Graphic Error Action (SGEA) data stream command for any characters otherwise unsupported.

The EBCDIC-to-ASCII translation tables provided with the Emulation program define the conversion of host EBCDIC characters to IBM Personal System/2 ASCII characters.

With the IBM 5216, the user must be sure that the correct printwheel is installed on the printer. The only characters supported are those on the printwheel. Refer to the printer operator's guide for more information about the characters on printwheels.

Note: If you are attaching a user-defined printer, it is your responsibility to understand the printer's limitations.

Forms Handling

The following table lists how the emulated host printers handle forms.

Printer Specification	IBM 5219	IBM 5224	IBM 5256
Print line length (inches)	13.2	13.2	13.2
Paper width: maximum continuous form (inches)	14.5	17.7	13.2
Paper width: maximum single sheet (inches)	15.4	14.5	14.5
Paper width: minimum continuous form (inches)	2.0	3.0	3.0
Paper length: maximum (inches)	14.33	14.0	14.0
Lines per inch (LPI)	See Note 1	6 or 8	6 or 8
Characters per inch (CPI) (see Note 2)	See Note 3	See Note 4	10
Characters per line (CPL)	See Note 5	132 198	132
Continuous forms	Yes	Yes	Yes
Single sheet forms	Yes	Yes	Yes

Notes:

1. The 5219 prints 4, 5-1/3, 6, 8, 9.6, 12, 24, or 48 LPI with 1, 1.5, 2, or 3 line distances between lines. This option is controlled by the host.
2. For compressed print, host printers use 15 CPI, while personal computer printers print 16 or 17 CPI, depending on the particular printer. Select the 15 CPI option when you want compressed print.
3. The 5219 prints 10, 12, 15, or proportional spaced CPI. This option is controlled by the host.
4. During 5224 emulation, 10 CPI or 15 CPI is controlled by the host.
5. The 5219 prints 132, 158, or 198 CPL (varies if proportional spacing is used). This option is controlled by the host.

Figure 3-1. Characteristics of Emulated Host Printers

EBCDIC-to-ASCII Translation

The following procedure allows you to view how EBCDIC characters are translated to ASCII on emulated printers.

1. Access the configuration program.
2. Select the advanced option, *Modify EBCDIC-to-ASCII Translation Table*. See Chapters 3 and 11 in the *User's Guide* for information about the configuration program and selecting options.
3. If you need a hard copy of the table, print the screen. See Chapter 4 in the *User's Guide* for information about printing the screen.

Note: The EBCDIC code point CA (hex 2D) is only used for IBM 5219 Printer emulation.

5219 Printer Commands

Figure 3-2 lists the 5219 commands and describes the level of support for each. Refer to the host functions reference manual for a description of these commands.

Hex Code	Command Description
03	<p><i>ASCII Transparent Data:</i> Printer emulation supports this command within a printer data stream, allowing a block of data to be transmitted from the host directly to the attached printer without conversion or inspection by the Emulation program.</p> <p>Hex 03 is a grouping control character that you use with a 1-byte parameter to denote the start of a transparent data stream. The parameter is a binary count of the bytes of transparent data (not including the count byte). The format of this command is "X'03' NN data..."</p> <p>Any value of the count parameter NN is valid. A count of 00 results in no operation. Emulation treats data following a 00 count in nontransparent mode.</p> <p>This command does not affect the internally maintained presentation position (PP). A command to clear the print buffers resets the ASCII transparent mode at any point within the transfer.</p>
09	<p><i>Superscript (SPS):</i> Some paper feed devices associated with some printers may not be capable of supporting superscript due to hardware limitations.</p>

Figure 3-2 (Part 1 of 7). 5219 Printer Commands

Hex Code	Command Description
1A	<i>Unit Backspace (UBS)</i> : This control is only supported on personal computer printers that have the ASCII Relative Move Inline Backward control. The IBM 5201 Quietwriter has this control, while the IBM 5216 Wheelprinter does not.
2BC2nnvv	<p><i>Set Vertical Format (SVF)</i>: The SVF control is converted to the ASCII control <i>Set Paper Length</i> (Esc-C) with the appropriate parameters and sent to the attached printer. A maximum presentation lines (MPL) parameter of 127 is supported by 5219 emulation. An MPL parameter value greater than 127 causes an EC = 2 condition.</p> <p>If a 5216 is attached, MPL divided by LPI must not exceed 15.</p> <p>If the 5216 is attached and an Esc-C command is sent to it on other than a page boundary, the 5216 generates an extra line feed.</p> <p>The Emulation program maintains the current line number to perform the auto page end function.</p>
2BC6...	<i>Set Line Density (SLD)</i> : This command is supported with all values of the line density (LD) parameter with the exception of 5.3 and 9.6 LPI. The Emulation program rounds down to the nearest supported value of LPI.

Figure 3-2 (Part 2 of 7). 5219 Printer Commands

Hex Code	Command Description
2BD10705...	<p><i>Set Font ID (FID) through Global Font ID (GFID):</i> 5219 emulation supports this command and all values for the GFID, FWD, and FA parameters. The Emulation program uses the same SFG table as the 5219 to determine the font style, pitch, and font width for a GFID received from the host. However, some of the fonts may not be available for the attached printer. The user must ensure that the font ID sent by the host is available on the attached printer.</p> <p>The Emulation program displays the hex value for each GFID received in the Output Data field on the emulated printer operator panel.</p> <p>When the first printable graphic following a valid SFG is encountered, the Emulation program sends the appropriate ASCII control codes to the attached printer, selects the desired font pitch, stops, and turns the Change Font indicator ON. The hexadecimal value identifying the GFID is displayed in the Output Data field. The user must then change the printwheel or font on the attached printer, make it ready, and then select the <i>Start</i> option on the printer operator panel. You can specify through configuration that the attached ASCII printer does not require user intervention to change fonts. Then, the Change Font indicator does not appear on the printer operator panel.</p>
2BD20309 1s	<i>Set Line Spacing (SLS):</i> This control is supported.
2BD2nn0D stpr	<i>Set Justify Mode (SJM):</i> This control is supported. The ASCII Set Horizontal Motion Index or Relative Horizontal Move Forward control is used to justify the right margin using the same rules as the 5219 Printer.

Figure 3-2 (Part 3 of 7). 5219 Printer Commands

Hex Code	Command Description																		
2BD20415 ldld	<i>Set Single Line Distance (SSLD)</i> : This control is supported with the exception of 5.3, 9.6, and 48 LPI and 1 line per centimeter. The Emulation program uses the next smaller supported value of LD if one of these values is received. If no smaller value exists, then the next larger supported value is used.																		
2BD20429 00cp	<p><i>Set Character Density (SCD)</i>: All values of the parameter for this command are supported. If no change to the current font width results from the SCD parameter value, the control is ignored.</p> <p>When the first printable graphic following a valid SCD is encountered, the Emulation program selects the proper pitch at the personal computer printer, stops, and turns the Change Font indicator ON. The user must change the printwheel or font on the attached printer, make it ready, and then select the <i>Start</i> option on the emulated printer operator panel.</p> <p>The GFID values displayed in the Output Data field for valid CD parameter values are:</p> <table><tr><th><i>CD Parameter (Hex)</i></th><th><i>Action Selected</i></th><th><i>GFID Value (Hex)</i></th></tr><tr><td>000A</td><td>10 CPI</td><td>0B</td></tr><tr><td>000B</td><td>Proportional spacing</td><td>9E</td></tr><tr><td>000C</td><td>12 CPI</td><td>57</td></tr><tr><td>000F</td><td>15 CPI</td><td>DF</td></tr><tr><td>00FF</td><td>10 CPI</td><td>0B</td></tr></table> <p>You can specify through configuration that the attached printer does not require user intervention to change fonts. Then, the Change Font indicator does not appear on the printer operator panel.</p>	<i>CD Parameter (Hex)</i>	<i>Action Selected</i>	<i>GFID Value (Hex)</i>	000A	10 CPI	0B	000B	Proportional spacing	9E	000C	12 CPI	57	000F	15 CPI	DF	00FF	10 CPI	0B
<i>CD Parameter (Hex)</i>	<i>Action Selected</i>	<i>GFID Value (Hex)</i>																	
000A	10 CPI	0B																	
000B	Proportional spacing	9E																	
000C	12 CPI	57																	
000F	15 CPI	DF																	
00FF	10 CPI	0B																	

Figure 3-2 (Part 4 of 7). 5219 Printer Commands

Hex Code	Command Description								
2BD2nn40 wdwddpdp	<i>Set Presentation Page Size (SPPS)</i> : This control is supported; however, the depth parameter is converted to lines per page and sent to the attached printer via the ASCII Esc-C (set page length) command. The width parameter is converted to column positions and maintained by the Emulation program. The width remains the same if the character pitch is changed.								
2BD2ln48 ...	<p><i>Page Presentation Mode (PPM)</i>: 5219 Printer emulation supports the Forms Control (FC), Source Drawer (SD), and quality (Q) parameters. No other parameters will be inspected.</p> <p>The 5219 Printer initializes paper feed technique to whatever is electrically connected to the printer. The Emulation program requires a description of paper feed hardware during configuration.</p> <p>If a source drawer is chosen by the print job that the personal computer printer is not capable of supporting, the paper feed technique (PFT) parameter of the Set Print Setup (SPSU) control will be overridden to force manual insertion. Printer emulation will display a Change Setup message with Output Data = 01 on the printer operator panel, indicating the change to manual insertion. Printer emulation will then display the Forms indicator with one of the following values in the Output Data field:</p> <table> <tr> <th>Value</th><th>Feed Action</th></tr> <tr> <td>01</td><td>from drawer 1</td></tr> <tr> <td>02</td><td>from drawer 2</td></tr> <tr> <td>0E</td><td>from the envelope hopper</td></tr> </table>	Value	Feed Action	01	from drawer 1	02	from drawer 2	0E	from the envelope hopper
Value	Feed Action								
01	from drawer 1								
02	from drawer 2								
0E	from the envelope hopper								

Figure 3-2 (Part 5 of 7). 5219 Printer Commands

Hex Code	Command Description
2BD2In48 ... continued	<p>If the PFT parameter of the SPSU control was overridden, a Change Setup message will indicate when the application returns to the PFT prior to the manual insertion override.</p> <p>If the PFT is continuous tractor and a change to the source drawer is requested, printer emulation will stop and illuminate Change Setup and display the new source drawer value in the Output Data field where D1 = Drawer 1, and D2 = Drawer 2.</p> <p>If you have defined a printer with a printer function table, the decision to override the PFT parameter of the SPSU control depends on what was specified during configuration and what functions were defined in the table. For example, to select and feed envelopes the user must answer yes to the envelope question during configuration, and also enter in the printer function table the ASCII control code sequence to select envelopes at the attached printer.</p>
2BD2nn49 tmtmbmbm	<p><i>Set Vertical Margins (SVM):</i> This control is supported except that the top margin parameter is rounded off to the nearest 1/48 inch.</p>

Figure 3-2 (Part 6 of 7). 5219 Printer Commands

Hex Code	Command Description												
2BD2nn4C xpxfxssi	<p><i>Set Printer Setup (SPSU):</i> When this control is received by 5219 emulation, the program stops, turns the Ready indicator OFF, turns the Change Setup indicator ON, and displays a code for the paper feed technique parameter received with this control. The codes displayed are:</p> <table><tr><th><i>PFT Value</i></th><th><i>Output Data Code</i></th><th><i>Feed Selection Description</i></th></tr><tr><td>01</td><td>01</td><td>Manual</td></tr><tr><td>02</td><td>02</td><td>Continuous tractor</td></tr><tr><td>03</td><td>03</td><td>Automatic cut sheet</td></tr></table> <p>The operator must manually change the paper feed method and then select the <i>Start</i> option on the printer operator panel. The 5219 Printer initializes paper feed technique to whatever is electrically connected to the printer. 5219 emulation requires the user to describe the paper feed hardware during configuration.</p>	<i>PFT Value</i>	<i>Output Data Code</i>	<i>Feed Selection Description</i>	01	01	Manual	02	02	Continuous tractor	03	03	Automatic cut sheet
<i>PFT Value</i>	<i>Output Data Code</i>	<i>Feed Selection Description</i>											
01	01	Manual											
02	02	Continuous tractor											
03	03	Automatic cut sheet											
38	<p><i>Subscript (SBS):</i> This command is supported. Some paper feed devices associated with some printers may not be capable of supporting subscript due to hardware limitations.</p>												
2BD10381...	<p><i>Set Coded Graphic Character Set (CGCS) through local ID (SCGL):</i> This control is not fully supported. 5219 emulation supports two character sets, one for use by data processing (DP) applications and one for word processing (WP) applications. If a valid SCGL command is received, the DP table will become the active table.</p>												
2BD10601...	<p><i>Set Graphic Character set Global registry ID (GCGID) through Global Character set ID (GCID) (SCG):</i> This control is not fully supported. If a valid SCG control with a Code Page Global ID (CPGID) parameter equal to decimal 256 is received, the WP table will become the active table; otherwise, the DP table will be used.</p>												

Figure 3-2 (Part 7 of 7). 5219 Printer Commands

5224 Printer Commands

The commands used by the host to communicate with the 5224 are completely documented in the host functional reference manuals. Some of the commands are listed here to explain the differences in the way they are supported with the Emulation program.

Commands Not Supported

The following commands are not supported by the Emulation program:

- Transparent (TRN)
- Load Alternate Character (LAC)
- Set Coded Graphic Character Set (CGCS) through local ID (SCL)

Note: Because the LAC command is not supported, the IBM System/36 5224 Advanced Printer Function licensed program is not supported.

Commands Supported

All other 5224 Printer commands are supported by the Emulation program. Figure 3-3 lists these commands and the level of support provided for each. Refer to the host functions reference manual for a complete description of these commands.

Hex Code	Command Description
03	<p><i>ASCII Transparent Data:</i> Printer emulation supports this command within a printer data stream, allowing a block of data to be transmitted from the host directly to the attached personal computer printer without conversion or inspection by the Emulation program.</p> <p>Hex 03 is a grouping control character that you use with a 1-byte parameter to denote the start of a transparent data stream. The parameter is a binary count of the bytes of transparent data (not including the count byte). The format of this command is "X'03' NN data..."</p> <p>Any value of the count parameter NN is valid. A count of 00 results in no operation. Printer emulation treats data following a 00 count in nontransparent mode.</p> <p>This command does not affect the internally maintained presentation position (PP).</p> <p>A command to clear the print buffers resets the ASCII transparent mode at any point within the transfer.</p>

Figure 3-3 (Part 1 of 3). 5224 Printer Commands

Hex Code	Command Description
2BC2...	<p><i>Set Vertical Format:</i> The SVF control is converted to the ASCII control <i>Set Paper Length</i> (Esc-C) with the appropriate parameters and sent to the attached printer. A maximum presentation lines (MPL) parameter of 127 is supported by 5224 emulation. An MPL parameter value greater than 127 causes an EC = 2 condition.</p> <p>If a 5216 is attached, the MPL divided by the lines per inch must not exceed 15 inches.</p> <p>If the 5216 is attached and an Esc-C command is sent to it on other than a page boundary, the 5216 generates an extra line feed.</p> <p>The Emulation program maintains the current line number to perform the auto page end function.</p>
2BC6...	<p><i>Set Line Density (SLD):</i> This command supports all line densities except 9 LPI. The Emulation program will round down to the nearest supported value of LPI.</p>

Figure 3-3 (Part 2 of 3). 5224 Printer Commands

Hex Code	Command Description												
2BD20429...	<p><i>Set Character Density (SCD):</i> All values of the parameter except 12 CPI are supported. If no change to the current font width results from the SCD parameter value, the control is ignored.</p> <p>When the first printable graphic following a valid SCD is encountered, the Emulation program selects the proper pitch at the attached printer, stops, and turns the Change Font indicator ON. The user must change the printwheel or font on the attached printer, make the printer ready, and then select the <i>Start</i> option on the emulated printer operator panel.</p> <p>The GFID values displayed in the Output Data field for valid CD parameter values are:</p> <table><tr><th><i>CD Parameter (Hexadecimal)</i></th><th><i>Action Selected</i></th><th><i>GFID Value (Hexadecimal)</i></th></tr><tr><td>000A</td><td>10 CPI</td><td>0B</td></tr><tr><td>000F</td><td>15 CPI</td><td>DF</td></tr><tr><td>00FF</td><td>10 CPI</td><td>0B</td></tr></table> <p>You can specify through configuration that the attached printer does not require user intervention to change fonts. Then, the Change Font indicator does not appear on the printer operator panel.</p>	<i>CD Parameter (Hexadecimal)</i>	<i>Action Selected</i>	<i>GFID Value (Hexadecimal)</i>	000A	10 CPI	0B	000F	15 CPI	DF	00FF	10 CPI	0B
<i>CD Parameter (Hexadecimal)</i>	<i>Action Selected</i>	<i>GFID Value (Hexadecimal)</i>											
000A	10 CPI	0B											
000F	15 CPI	DF											
00FF	10 CPI	0B											

Figure 3-3 (Part 3 of 3). 5224 Printer Commands

Error Conditions

The following error conditions can occur during printer emulation.

Condition	Indication
End of Forms	PE line went active in the parallel interface.
Printer Not Ready	Error line went active with the PE line inactive in the parallel interface.

Chapter 4. Using the Printer Function Table Setup Program

Introduction	4-2
PFTs Provided by the Emulation Program	4-2
Creating a Printer Function Table	4-3
Entering Control Sequences	4-4
Variable N	4-5
Determining the Format of Variable n	4-6
Defining Printer Functions	4-6
Initialization and Reset	4-7
Start of Document Initialization	4-7
End of Document Reset	4-8
Vertical and Horizontal Spacing	4-8
Vertical Line Spacing	4-8
Horizontal Character Spacing	4-9
Horizontal Relative Movement	4-10
Proportional Space Mode and Justification	4-10
Highlighting	4-11
Paper and Page Options	4-11
Paper Handling	4-11
Page Length	4-12
Type Styles and Character Sets	4-13
Default Type Style	4-13
Individual Type style	4-13
Group Type Styles	4-14
Character Set Redefinition	4-14
Function Selection Tests	4-15

Introduction

The Printer Function Table (PFT) describes the functional characteristics of a personal computer printer to the Emulation program. You create and edit this table with the setup program. If your printer is not supported, and cannot function as a supported printer, you need a PFT to use the printer with the Emulation program. See "Requirements for IBM Personal System/2" in Chapter 1 of the *User's Guide* for a list of supported printers.

PFTWS.PRN is a worksheet for the program. You can print this file and use it when preparing for the program.

Note: The Emulation program may not require or use all the functions provided by the printer. Also, the Emulation program does not use all the functions provided by the setup program itself. Only those functions described in this chapter are supported by the Emulation program.

PFTs Provided by the Emulation Program

The Emulation program provides the following tables:

DEFAULT.PFT

This table allows printing of simple output (no changes in pitch, type style, and so on). You may be able to use this table initially to define your printer. If you need to create a PFT for more complex functions, you can copy this PFT and modify it as needed.

IBM4202.PFT

This PFT supports the IBM 4202 Proprinter XL.

IBM3812.PFT

This PFT supports the IBM 3812 Pageprinter. If you have a 3812, you must use this PFT. You cannot completely recreate this table with the setup program. Do not modify the Type Styles and Character Sets portion of this table.

Creating a Printer Function Table

Use this procedure to create a PFT:

1. If you have a serial printer, use the DOS MODE command to direct the printer output to the serial port before you run the PFT setup program. At the DOS prompt type the following commands, pressing Enter at the end of each line.

```
MODE COM1:1200,n,8,1,p
```

```
MODE LPT1:=COM1
```

The baud rate and other parameters on the COM1 line vary depending on the printer. See the printer operator's guide for the correct information to enter on the COM1 line.

2. To start the program, type PFTSETUP and press Enter.
3. Select the option to create a table.
4. A screen of information appears. Read the information, then press Enter to continue.
5. Enter a PFT filename. Use the second option if you need to make changes to a PFT you have created, or a PFT provided by the Emulation program.
6. The Revise Printer Function Table menu appears. Choose the option you want to define.
7. The program prompts you for all necessary information about printer functions. See "Defining Printer Functions" on page 4-6 for a description of each printer function. See "Entering Control Sequences" on page 4-4 for information on entering control sequences in the PFT.

Refer to the printer operator's guide for the correct terminology for each function. Study the functions and the concepts described to determine the appropriate use or need for each PFT item.

8. After you finish defining functions, run the Function Selection Tests (see "Function Selection Tests" on page 4-15).
9. Update the session profile to use the PFT (see Chapter 3 in the *User's Guide*).

Entering Control Sequences

A control sequence is a sequence of hex values that instruct the printer to perform a specific function or print a particular output. Use these rules when entering control sequences:

- Use two hex digits for each byte to be sent to the printer.
- Leave one space between each pair of hex digits.
- Use either upper or lowercase letters.
- Enter a maximum of 41 characters, including spaces, in the first field of a control sequence (14 bytes of hex values).
- Enter a maximum of 59 characters, including spaces, in the continuation field (20 bytes of hex values).
- Use a variable placeholder (*n* or *N*) separated by a space, if a printing function can be changed at print time by specifying a variable to the printer.
- Multiple controls can be used in one control sequence. However, you can only use one *variable n* per sequence.
- Do not enter a *variable n* as the first byte of a control sequence.

For example, if the printer operator's guide specifies Esc-E for a particular function, type 1B 45 (1B is the ASCII hex value for Esc, and 45 is the value for E).

Note: If a control sequence sets a printer mode that affects the operation of other control sequences, be sure to reset this mode in the affected sequences.

Variable N

Each printer interprets the value for *variable n* differently. The interpretation is based on this algorithm:

$$(\text{value sent to printer} + / - \text{offset}) * x/y = n$$

where:

Maximum value determines the type of encoding and number of bytes to send to the printer (the *value*, or *variable n*, in the algorithm). See the printer operator's guide for the maximum decimal value allowed for each *variable n*. For example, if the printer operator's guide defines a control in which a value of zero is equal to a parameter value of hex 40 (decimal 64), and a maximum number of units supported by the control is hex 0F (decimal 15), then the maximum value of *n* is 79. Or, if the printer operator's guide defines a control sequence, such as Esc-US-*n*, where *n*-1 is the value used by the printer and the value of *n* cannot exceed hex 7E (decimal 126), then the maximum value of *n* is 126.

Offset added or subtracted by printer provides a way to enter a decimal number that the printer adds or subtracts to the *variable n* when it receives the control sequence. For some printers, the value of *n* must be adjusted before being sent to the printer, because the printer performs an adjustment on the value. For example, if the printer operator's guide states *n* + 1 is the value used by the printer, then the offset is 1 and is added to the variable. Or, if the printer operator's guide specifies a chart where a value of zero equates to a parameter of hex 40 (decimal 64), then the offset is 64, which is subtracted from the variable.

Control Increments (x/y inches) represent a scaling factor used to convert the value of *variable n* to inches. For some printers, the value of *n* must be defined in fractions of an inch. For example, if $x/y = 1/72$, the variable represents units of 1/72 inch.

When $x/y = 1/1$, the variable represents units of 1 inch, and the algorithm is simplified to this:

$$(\text{value sent to the printer} \pm \text{offset}) = n$$

If the printer operator's guide defines n (or n times a fraction) as the value used by the printer (for example, $n/72$), the offset is zero. The algorithm simplifies to this:

$$(\text{value sent to the printer} * x/y) = n$$

Determining the Format of Variable n

The Emulation program requires that you run the test to determine the format of *variable n* .

Each *variable n* must be in a specific format for the printer. The PFT setup program helps you specify the format. When you press Enter in a menu that has just been modified and that defines a *variable n* , the PFT setup program tests *variable n* on the printer. You then examine printed output and respond to questions on the screen. These responses determine the value of *variable n* . If necessary, you can redefine the control sequence and press Enter again to determine the type of variable. If none of the tests provides the correct result, the Emulation program cannot use the function on your printer.

Defining Printer Functions

The Revise Printer Function Table menu lists the control sequences you can define for printer functions:

- Initialization and reset
- Vertical and horizontal spacing
- Highlighting
- Paper and page options
- Type styles and character sets
- User-defined controls

The Emulation program does not support user-defined controls.

Initialization and Reset

The PFT setup program allows you to define Start of Document Initialization and End of Document Reset.

Start of Document Initialization

The Emulation program uses this control when a printer session is first started or when a printer session is resumed after having been suspended. You may define a control in the Start of Document control sequence or set a printer switch to set each of the following conditions.

Note: The PFT setup program sends the control sequence for Start of Document Initialization to the printer before every test.

Vertical Units	Units defined for the Variable Line Spacing control.
Line Spacing	6 lines per inch.
Pitch	10 pitch.
Carriage Return	With no line feed. If a carriage return cannot be achieved without a line feed, zero index carriage return (ZICR) is not supported. Also, line spacing may always be incorrect if the sequence carriage return line feed causes an extra line feed.
Line Feed	Perform a line feed with no carriage return.
Turn Off	Superscript, subscript, underline, overstrike, bold, PSM, justification.
Character Set	Default of the printer. Set to PC character set 2 if the printer supports it.
Printing Mode	Bidirectional or unidirectional mode depending on the printer.
Color	Black.
Line Length	Maximum supported by the printer.
Page Length	11 inches.

Do not define a control sequence that can affect a preceding print job.

End of Document Reset

The Emulation program uses the control sequence for End of Document Reset when a printer session is suspended. You should define a control (where applicable) in this control sequence to achieve the following conditions:

- Vertical units
- Line spacing (in default vertical units)
- Carriage return status
- Line feed status
- Page length
- Left margin
- Printer font (pitch and type style)
- Printer character set
- Portrait and landscape status
- Color
- Printing mode

Vertical and Horizontal Spacing

The Vertical and Horizontal Spacing option allows you to define these functions:

- Vertical line spacing
- Horizontal character spacing
- Horizontal relative movement

Vertical Line Spacing

Vertical line spacing is a measure of the distance between printed lines. You can define vertical line spacing by lines per inch (LPI) or variable line spacing (VLS).

The Emulation program uses the controls for 6 or 8 LPI. Otherwise, the VLS control is used. VLS must be defined to support 4, 12, and 24 LPI. If your printer has specific controls for 6 and 8 LPI, you should define all three sequences.

Lines per Inch: The most common vertical line spacing is 6 or 8 LPI. The Emulation program supports 4, 6, 8, 12, and 24 LPI. You can define a control sequence to set 6 LPI and one to set 8 LPI.

Variable Line Spacing: A VLS control sets the vertical line spacing to a variable value. Some printer manuals refer to VLS as Line Feed Pitch. Check the printer operator's guide, especially if a variable increment is allowed, for the function that controls vertical motion.

If you define a VLS, try multiples of 1/48 first. Otherwise, use the smallest increment of vertical spacing supported by the printer. Common values are 1/48, 1/72, and 1/96.

Indexing Functions: The Emulation program uses indexing to support superscripts and subscripts.

The Begin and End Superscript and Subscript controls function on all feed devices. On some printers, these controls move the print position up or down 1/2 line spaces. Other printers create superscripts and subscripts by compressing the height of the characters. If your printer can only print superscripts and subscripts in one pitch, do not define these controls.

The Emulation program does not support 1/2 Index Up (reverse), 1/2 Index Down (forward), or Reverse Index. These controls move the print position up or down 1/2 the active line spacing.

Horizontal Character Spacing

Horizontal character spacing refers to the number of characters in a horizontal inch of printing. You can set up Horizontal Character Spacing as either a pitch value or with Horizontal Motion Index (HMI). The value of a pitch is determined by the width of the characters and how far apart they are spaced on the printed page. For example, 10 pitch refers to 10 characters in a horizontal inch. The Emulation program supports 10, 12, 15, and 17 pitches. It does not support 5 and 8.55 pitches. PSM is supported with the HMI control.

Some printers use the HMI control to support pitch. HMI causes the print element to move the appropriate increment to position the characters. If you define an HMI, try multiples of 1/60 inch. If you cannot use these multiples, use the smallest spacing supported by the printer. Common values are 1/60 and 1/120 inch.

For each pitch in a print job, the Emulation program first checks for a defined pitch control sequence. If it is defined, the active pitch is set accordingly. Otherwise, the Emulation program sets the defined pitch control nearest to the one received. If no pitch control sequence is defined, the Emulation program prompts the user to change the physical font at the attached printer.

HMI is also used to justify text and to support proportionally spaced fonts.

Horizontal Relative Movement

The Horizontal Relative Movement (HRM) controls move the print position backward or forward relative to the current print position. Some printer manuals refer to HRM as Horizontal Cursor Positioning. Find the function in the printer operator's guide that moves the horizontal position of the printer relative to the current position.

If possible, define control sequences for forward and backward relative movement. Try multiples of 1/60 inch first. Otherwise, use the smallest increment of horizontal spacing supported by the printer. Common values are 1/120 and 1/240 inch.

The Forward Relative Movement control can be used to justify text. The Backward Relative Movement control can be used to unit backspace.

Proportional Space Mode and Justification

You must define HMI or HRM controls to space words evenly on a justified line. You must define HMI to space text proportionally. Otherwise, PSM is not supported.

For 5219 emulation, you must define either HMI or forward HRM. If you want to use unit backspacing with 5219 emulation, you must define backward HRM.

Highlighting

The Emulation program supports Begin Quality Print and End Quality Print.

The Emulation program does not support Begin Emphasis and End Emphasis controls. The host uses backspace and overstrike to print bold in 5219 emulation. The Emulation program does not support Begin Underline and End Underline. Multiple passes on the print line are used to underline text.

Paper and Page Options

The paper sources defined during configuration can be supported by defining paper handling and page length controls in the PFT setup program. The Emulation program does not support the Paper Positioning, Set Left Margin, Carriage Return and Line Feed, Multiple Copies, or Landscape Paper Handling controls. If your printer does not have advanced paper handling capabilities, the Emulation program provides default values for the controls that are not specifically defined.

Paper Handling

To control the paper source requested by a print job, the Emulation program uses the control sequences for Bottom, Top, and Envelope Tray Select and Feed for paper feed; Manual Feed Select; and the Automatic Cut Sheet option for ejecting sheets without paper feed. The Emulation program does not support the controls for Continuous Feed Select, Ignore Paper End Sensor, Enable Paper End Sensor, Manual Cut Sheet, and Collate.

Some printers require two controls for the paper (or envelope) feed function: one for paper tray selection and one for loading paper from that tray. In this case, define both controls in the control sequence.

If the printer has an automatic sheet-feed device, define all five paper feed items if possible. If the printer supports a paper feed such as single-drawer automatic sheet feed, but does not define a control sequence to select or feed paper from that paper feed, define a null (hex 00) for that paper feed.

The select and feed controls in the Paper Handling menu can be defined for the bottom, top, and envelope tray. This control causes a sheet of cut paper (or envelope) to be fed into the printer from the specified tray. If the print job indicates that one of these trays is to be used, and no control is defined, the Emulation program uses the manual feed select control.

The manual feed select control allows the automatic sheet-feed device to accept a manually inserted sheet of paper through the manual insertion gate. The printer must stop printing to accept the manually fed sheet.

The Emulation program assumes that continuous forms feed is available.

If the eject sheet without paper feed (automatic cut sheet) control is defined, the Emulation program assumes the paper feed controls eject the current page and feed the next page.

The Emulation program assumes the manual feed select control ejects the current page and stops the printer for the next manually inserted page.

Run the print test for continuous feed select on a continuous forms feed. Run the print tests for bottom, top, and envelope trays on an automatic feed. Run the print test for manual paper feed insertion on a manual feed or automatic feed with a manual insertion gate.

Page Length

The Emulation program uses the page length controls to set the length of the page. You must specify the page length controls with one of the options (inches or lines).

Type Styles and Character Sets

Type style defines the appearance (font) and pitch of characters. For example, Gothic characters look different from characters printed in a commonly used type style such as Prestige Elite. You can define default, individual, and group type styles.

Character sets are defined groups of graphic characters. For example, a character set can include a through z, A through Z, and other print characters.

For each individual and group type style definition, you can select a PC character set. When you print, all type styles not defined use the default type style definition. The default definition selects a PC character set:

If the printer has removable print elements such as a printwheel, you can define a character set to match those on a print element, then associate the character set with a type style number.

You can also define a character set to include characters that can be constructed from those on the print element. For example, if the print element has a tilde, you can define the *n tilde* character as *n backspace tilde*.

The Emulation program does not support Slot Selection Definition.

Default Type Style

Type style definitions allow you to specify the controls that select the type styles and characters sets supported by your printer. Type styles are defined with initial and ending control sequences for PC character sets 2 through 9. The initial control sequence selects the print element for a type style and a character set. The ending control sequence is optional. The Emulation program outputs this sequence before selecting a new type style or character set.

Individual Type style

An individual type style definition can be defined to select a type style and associated character sets. Use an individual definition to define a type style that can only be selected using unique control sequences for each print element.

Group Type Styles

You can define a Group Type Style to select a group of type styles and associated character sets on the printer. Each group type style definition is assigned a group type style identifier. There are 26 identifiers (A through Z). Use a group type style definition to define a group of type styles that are selected through common control sequences. For example, you can define a group type style definition for a print element that supports the PC character set and Courier type style in 10, 12, and 15 pitch. The corresponding type style numbers you would type in the definition would be 11, 85, and 223.

Character Set Redefinition

The Emulation program supports PC character sets 2 through 9.

Function Selection Tests

After you create or revise a PFT, run the function selection tests to verify the print functions you defined in the table:

- Superscript and subscript
- Form feed
- Pitches
- Backspace

You do not need to run the tests for the functions not supported by the Emulation program (Underline, Emphasis, and Cursor Draw).

Use this procedure to run the tests:

1. Choose the Function Selection Test option from the main menu.
2. Select a test and prepare the printer as directed by the test.
3. The test will print samples. Answer the questions on the screen according to the output from the printer. When more than one example is printed, choose the first one that provides the appearance you want.
4. If none of the printed results appear as you want, choose the example that is closest. Or, you can redefine the control sequence for the function.
5. If you decide to redefine any controls, run the tests again.

Note: If a control that is not defined is used in a test, the test is printed without that control.

Run the tests to ensure that you have defined the print functions correctly. If you do not, the PFT setup program prompts you to run them before you exit the program.

Section 100

100.1. The purpose of this section is to provide for the orderly and efficient operation of the court system.

100.2. The court system shall be organized as follows:

(a) The court system shall be organized as follows:

(b) The court system shall be organized as follows:

(c) The court system shall be organized as follows:

100.3. The court system shall be organized as follows:

100.4. The court system shall be organized as follows:

100.5. The court system shall be organized as follows:

100.6. The court system shall be organized as follows:

100.7. The court system shall be organized as follows:

100.8. The court system shall be organized as follows:

100.9. The court system shall be organized as follows:

100.10. The court system shall be organized as follows:

100.11. The court system shall be organized as follows:

100.12. The court system shall be organized as follows:

100.13. The court system shall be organized as follows:

100.14. The court system shall be organized as follows:

100.15. The court system shall be organized as follows:

100.16. The court system shall be organized as follows:

100.17. The court system shall be organized as follows:

100.18. The court system shall be organized as follows:

100.19. The court system shall be organized as follows:

Chapter 5. Graphics Data Stream

Introduction	5-2
Fully Supported and No-op Orders	5-4
Modified Orders	5-4
Graphics Control Orders	5-5
Graphics Display On Order (93)	5-6
Graphics Display Off Order (94)	5-7
Graphics Set Orders	5-8
Set Color Table Order (B4)	5-8
Set Color Order (B0)	5-11
Set Style Order (B1)	5-13
Set Function Order (B3)	5-15
Set Fill Mode Order (B7)	5-17
Set Attributes (B8)	5-20
Graphics Draw Orders	5-21
Write Background Order (A3)	5-22
Read Orders	5-23
Read Status Order (80)	5-23
Read Attributes Order (81)	5-26
Printer Orders	5-28
IEEE Interface Orders	5-29
IEEE Data Follows Order (E0)	5-30
Take Control Asynchronously Order (E1)	5-31
Go To Standby Order (E2)	5-32
Interface Clear Order (E5)	5-32

Introduction

The orders used by the host to communicate with the Emulation program in graphics mode are completely documented in the *IBM 5250 Information Display System Functions Reference Manual*. Refer to this manual to make any changes to the data stream. All orders are listed here; however, only the orders that are used in a different way are described.

The following is a list of the graphics orders.

Order Name	Graphic Byte (Hex)	Program Action
Read Status	80	Modified
Read Attributes	81	New order
End Graphics Block	90	Fully supported
More Data to Come	91	Fully supported
End of Data	92	Fully supported
Graphics Display On	93	Modified
Graphics Display Off	94	Modified
End Graphics	95	Fully supported
Suppress Pacing Response	96	Fully supported
Draw Polyline	A0	Fully supported
Draw Scanline	A1	Fully supported
Write Background	A3	Modified
Write Polymarker	A4	Fully supported
Fill Polygon	A5	Fully supported
Define Shield Area	A6	No-op order
Set Color	B0	Modified
Set Style	B1	Modified
Set Style Offset	B2	No-op order

Figure 5-1 (Part 1 of 2). Graphics Orders

Order Name	Graphic Byte (Hex)	Program Action
Set Function	B3	Modified
Set Color Table	B4	Modified
Set Marker	B5	Fully supported
Set Line Weight	B6	No-op order
Set Fill Mode	B7	Modified
Set Attributes	B8	New order
Printer Data Follows	C0	P1 error (printer not attached)
Screen Copy	C1	P1 error (printer not attached)
Load Printer A/N Color Mix Table	C2	No-op order
Load Printer Graphics Mix Table	C3	No-op order
Set Printer Time-out	C4	No-op order
Set Address	D1	Modified
Set Talker Time-out	D2	Modified
Set EOI Mode	D3	No-op order
IEEE Data Follows	E0	Modified
Take Control Asynchronously	E1	Modified
Go To Standby	E2	Modified
Remote Enable	E3	No-op order
Remote Disable	E4	No-op order
Interface Clear	E5	Modified

Figure 5-1 (Part 2 of 2). Graphics Orders

Fully Supported and No-op Orders

CAUTION

The IBM Personal System/2 operates at speeds unrelated to the 5292-2 making operation with the *suppress pacing response* order unpredictable.

Fully supported and no-op orders are listed in Figure 5-1 on page 5-2. If you need to know more about these orders, refer to the *IBM 5250 Information Display System Functions Reference Manual*.

All no-op orders are checked for correct 5292-2 format and context. A message appears if an error is detected.

Modified Orders

Modified orders include the following:

- Graphics control orders
- Graphics set orders
- Graphics draw orders
- Read orders
- Printer orders
- IEEE interface orders

The description of each modified graphics order follows. The hex value of each graphics order is given after the name of the order.

Graphics Control Orders

Order Name	Graphics Byte (Hex)	Described
End Graphics Block	90	No
More Data to Come	91	No
End of Data	92	No
Graphics Display On	93	Yes
Graphics Display Off	94	Yes
End Graphics	95	No
Suppress Pacing Response	96	No

Figure 5-2. Graphics Control Orders

Graphics Display On Order (93)

The *graphics display on order* turns on the graphics display. This allows the display station to display data from the graphics buffer. Graphics display is on when **G** (in reverse image) is shown on the status line.

The Emulation program enters the requested graphics state, sets the resolution, clears the screen, and redraws any data in the graphics buffer using the orders in the graphics buffer. Any text that was on the screen at the time the order was received is then added to the display.

Notes:

1. Shifting into and out of graphics state (by using the Hot Key function to leave and return to the session, or with the Local Select mode 1 command) may cause the size and exact location of text to change in relationship to graphics data.
2. When operating in 320 by 200 pixel mode all text is displayed in 40 characters per row. Text beyond column 40 is not displayed.

Format

Byte 1							
Graphic Order							
9				3			
1	0	0	1	0	0	1	1

RB3F001

Graphics Display Off Order (94)

The *graphics display off* order turns off the graphics display. Graphics display is off when a G is shown (in normal image) on the status line.

The graphics state is reset and the program returns to the previous text state. Any data in the text buffer is displayed using the current screen size and attributes. All graphics data in the VDI buffer is saved until the next *write background* order or the Local Select mode Erase key is pressed.

Format

Byte 1							
Graphic Order							
9				4			
1	0	0	1	0	1	0	0

RB3F002

Graphics Set Orders

Order Name	Graphics Byte (Hex)	Described
Set Color	B0	Yes
Set Style	B1	Yes
Set Style Offset	B2	No
Set Function	B3	Yes
Set Color Table	B4	Yes
Set Marker	B5	No
Set Line Weight	B6	No
Set Fill Mode	B7	Yes

Set Color Table Order (B4)

The *set color table* order loads the display's color table with new color intensity values. Each 9-bit entry consists of three intensity bits for each primary color (red, green, and blue). This allows for intensities of 0 through 7 for each color. When supported by the graphics adapter, it provides up to 512 possible colors or shades. A value of 7 indicates the maximum intensity for a primary color.

When display hardware supports a color table, the command is fully implemented but the actual color depends on the display hardware capabilities. This command is accepted but ignored when an IBM Personal System/2 display does not support a color table.

If the display hardware supports fewer than eight colors but has color table support, you can map the eight colors from the host to the color of the IBM Personal System/2 hardware through the graphics configuration program. See Chapter 11 in the *User's Guide* for more information.

The display's color table has the following default values:

3-Bit Index	Color (RRR GGG BBB)	Default Color Displayed
0	000 000 000	Black
1	111 000 000	Red
2	000 111 000	Green
3	000 000 111	Blue
4	111 000 111	Pink
5	111 111 000	Yellow
6	000 111 111	Turquoise
7	111 111 111	White

Format

Byte 1 Graphic Order								Byte 2 Graphic Data							
B				4				4				x			
1	0	1	1	0	1	0	0	0	1	0	0	0	n	n	n

Byte 3 Graphic Data								Byte 4 Graphic Data							
x				x				x				x			
0	1	r	r	r	g	g	g	0	1	b	b	b	0	0	0

RB3F003

Byte	Field Content	Field Description
1	Hex B4	Order code.
2	Table index (low 3 bits)	Index into display's color table.
3, 4	New table value	<p>New value for display's color table entry specified by table index.</p> <p>Data byte format for entry change:</p> <p>01RRRGGG R = red intensity G = green intensity</p> <p>01BBB000 B = blue intensity</p>
5-n	...	Additional index/value bytes.
n + 1	Hex 92	End of data order.

Applicable Draw Orders: The following draw orders are applicable to this order.

Draw Polyline
 Draw Scanline
 Fill Polygon
 Write Background
 Write Polymarker

Set Color Order (B0)

The *set color* order specifies the color index to be used for subsequent draw orders.

The default color index is 7.

Format

Byte 1								Byte 2							
Graphic Order								Graphic Data							
B				0				4				0-7			
1	0	1	1	0	0	0	0	0	1	0	0	0	a	a	a

RB3F004

Byte	Field Content	Field Description
1	Hex B0	Order code.
2	Graphics data	The binary value of bits aaa specify the color index (0 through 7) to be used for subsequent draw orders.

Applicable Draw Orders: The following draw orders are applicable to this order.

Draw Polyline
Draw Scanline
Fill Polygon
Write Polymarker

Example: The following graphics order and data byte set the display color to color index 3 for subsequent draw orders:

Byte 1								Byte 2							
Graphic Order								Graphic Data							
B				0				4				3			
1	0	1	1	0	0	0	0	0	1	0	0	0	0	1	1

RB3F005

Note: The color value is used as an index into a table of selected colors, which is supplied as a part of the Emulation program. This color applies to all subsequent graphics objects except text (which gets its color from its field attributes).

Set Style Order (B1)

The *set style* order specifies the line style to be used for subsequent draw orders.

The default is a solid line.

Format

Byte 1 Graphic Order	Byte 2 Graphic Data	Byte 3 Graphic Data
B	4	x
1 0 1 1 0 0 0 1	0 1 0 0 a a a a	0 1 0 0 b b b b

Byte 4 Graphic Data	Byte 5 Graphic Data
4	x
0 1 0 0 c c c c	0 1 0 0 d d d d

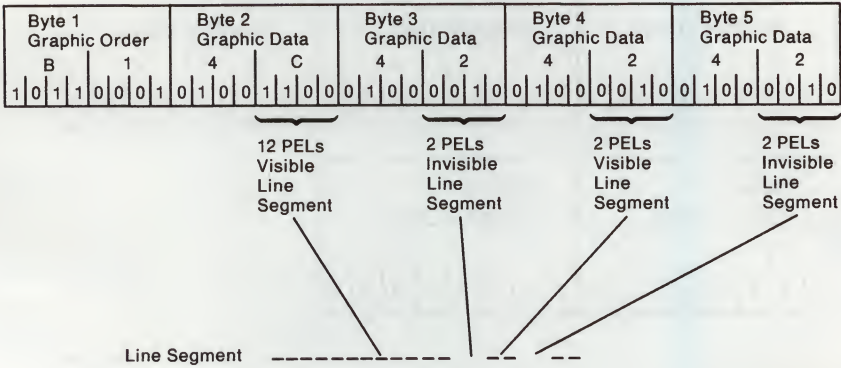
RB3F006

Byte	Field Content	Field Description
1	B1	Order code.
2	Segment A	Length (binary value of the aaaa bits; number of PELs) of the first visible segment of the line style.
3	Segment B	Length of the first unchanged segment (binary value of the bbbb bits; number of PELs) that follow the first visible segment.
4	Segment C	Length of the second visible segment (binary value of the cccc bits; number of PELs) of the line.
5	Segment D	Length of the second unchanged segment (binary value of the dddd bits; number of PELs) that follow the second visible segment.

Applicable Draw Orders: The following draw orders are applicable to this order.

Draw Polyline
Fill Polygon

Example: The following graphics order and graphics data set the line style as shown below on a blank background.



RB3F043

IBM Personal System/2 Mapping: The VDI line style is set according to the segment specifications described above and the selections made during graphics configuration. The supported VDI line styles are shown in the following table.

Style	Description
0	Invisible
1	Solid
2	Long dash
3	Dotted
4	Dash dotted
5	Medium dash
6	Dash with two dots
7	Short dash

Set Function Order (B3)

The *set function* order determines how subsequent draw orders are combined with the existing picture (writing mode). This order allows you to specify whether you draw over an existing picture (replace), or combine the existing picture color with the color you are drawing to display a different color.

The default is to replace the existing color with the color you are drawing.

Format

Byte 1 Graphic Order							Byte 2 Graphic Data						
B				3			4				x		
1	0	1	1	0	0	1	1	0	1	0	0	0	a

RB3F008

Byte	Field Content	Field Description
1	Hex B3	Order code.
2	Function (low 2 bits)	Drawing logical operator. 00 Ignored by the IBM Personal System/2. 01 OR: The color you are drawing is combined with the existing color using OR logic. 10 XOR: The color you are drawing combines with the existing color using exclusive OR logic. 11 REPLACE: The color you are drawing replaces the existing color.

Applicable Draw Orders: The following draw orders are applicable to this order.

- Draw Polyline
- Draw Scanline
- Fill Polygon

Example: If you are drawing a red line and encounter an existing white PEL, the resulting color (assuming default colors) is:

OR	0 1 0	(Color index 2)
	1 1 0	(Color index 6)

	1 1 0	Resulting color index

Order Used	Byte 1								Byte 2							
	Graphic Order								Graphic Data							
	B				3				4				1			
	1	0	1	1	0	0	1	1	0	1	0	0	0	0	0	1

RB3F009

IBM Personal System/2 Mapping: The virtual device interface supports the same writing modes as the 5292-2, but handles the writing of styled lines differently. See Chapter 6, "Graphics Display."

Set Fill Mode Order (B7)

The *set fill mode* order establishes the fill mode control information that is used with the *fill polygon* order.

The default values are as follows:

Bits	Value	Description
aa	00	Ignored.
bb	00	Draw a solid edge and fill interior with style.
ccccc	000000	Ignored.

Format

Byte 1 Graphic Order								Byte 2 Graphic Data								Byte 3 Graphic Data							
B				7				4				x				x							
1	0	1	1	0	1	1	1	0	1	0	0	a	a	b	b	0	1	c	c	c	c	c	c

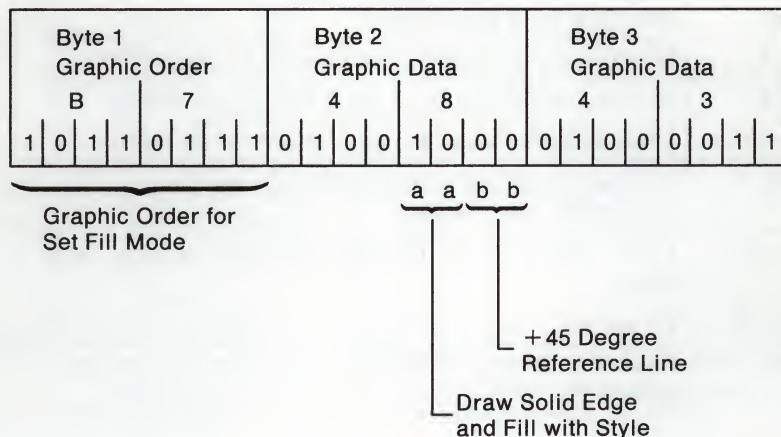
RB3F010

Byte	Field Content	Field Description
1	Hex B7	Order code.
2, 3	Fill control	<p>Control information for area fill.</p> <p>bb bits Ignore.</p> <p>aa bits 00 Draw solid boundary line and fill with style specified in <i>set style</i> order.</p> <p>aa bits 01 Draw solid boundary line.</p> <p>aa bits 10 Fill interior with style specified in <i>set style</i> order.</p> <p>Note: Fill interior includes the edge.</p> <p>aa bits 11 Draw boundary with style specified in <i>set style</i> order.</p>

Applicable Draw Orders: The following draw order is applicable to this order.

Fill Polygon

Example: The following graphics order and graphics data result in drawing a solid boundary line and filling the interior with the style defined by the *set style* order in effect when the *fill polygon* order is issued.



RB3F011

IBM Personal System/2 Mapping: Control byte 1, bb bits and control byte 2, ccccc bits are ignored. A VDI fill pattern, if requested, is based on the current VDI line style.

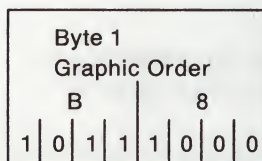
Line Style	Fill Pattern
0	Hollow
1	Solid
2	Wide spaced 45° lines crossed with -45° lines
3	Wide spaced 45° lines
4	Medium spaced 45° lines
5	Medium spaced 45° lines crossed with -45° lines
6	Narrow spaced 45° lines
7	Narrow spaced 45° lines crossed with -45° lines

Set Attributes (B8)

The *set attributes* order is followed by 40 bytes of data, which determine the following:

Bytes	Attribute Data Description
1 to 4	Number of horizontal units used (0 through 32767).
5 to 8	Number of vertical units used (0 through 32767).
9 to 12	Graphic buffer size (in bytes, 256 through 1920).
13 to 16	Number of screen colors (2 through device maximum). See Note.
17 to 20	Color palette size (0 through device maximum). See Note.
21 to 40	Reserved.
Note: The values for screen color and palette size must not be changed by the host.	

Format: The first byte has this format:



RB4L037

The 40 bytes following the order have the format **01xxxxxx**, where **x** is an offset between 0 and 1920 (offset 0 corresponds to row 1, column 1), and vary as stated above.

For example, if the horizontal resolution of the graphics display is 480, then the maximum vertical resolution is 479 ($479 = 480 - 1$), which is hex 1DF. Bytes 5 through 8 of the attribute information would appear as follows:

<p>Byte 5 Graphics Data</p> <p>4 0</p> <p>0 1 0 0 0 0 0 0</p>	<p>Byte 6 Graphics Data</p> <p>4 0</p> <p>0 1 0 0 0 0 0 0</p>
<p>Byte 7 Graphics Data</p> <p>4 7</p> <p>0 1 0 0 0 1 1 1</p>	<p>Byte 8 Graphics Data</p> <p>5 F</p> <p>0 1 0 1 1 1 1 1</p>

RB4L038

Graphics Draw Orders

Order Name	Graphics Byte (Hex)	Described
Write Background	A3	Yes
Draw Polyline	A0	No
Draw Scanline	A1	No
Write Polymarker	A4	No
Fill Polygon	A5	No
Define Shield Area	A6	No

Write Background Order (A3)

The *write background* order erases the graphics display using the specified background color.

Format

Byte 1 Graphic Order								Byte 2 Graphic Data							
A				3				4				x			
1	0	1	0	0	0	1	1	0	1	0	0	0	a	a	a

RB3F012

Byte	Field Content	Field Description
1	Hex A3	Order code.
2	Background color	The binary value of the aaa bits is the color index this order uses when it writes the background.

Applicable Draw Orders: The following draw order is applicable to this order.

Set Color Table

Example: The entire graphics display is changed to the color indicated by an index of 1 in the color table.

Byte 1 Graphic Order								Byte 2 Graphic Data							
A				3				4				6			
1	0	1	0	0	0	1	1	0	1	0	0	0	1	1	0

RB3F013

The color specified by the color index of the host color table covers the entire graphics area of the screen.

Read Orders

Order Name	Graphics Byte (Hex)	Described
Read Status	80	Yes
Read Attributes	81	Yes

Read Status Order (80)

The *read status* order prepares graphics status information to be read by the host. The information is stored in the alphanumeric display buffer at the offset specified by the two bytes that follow the order.

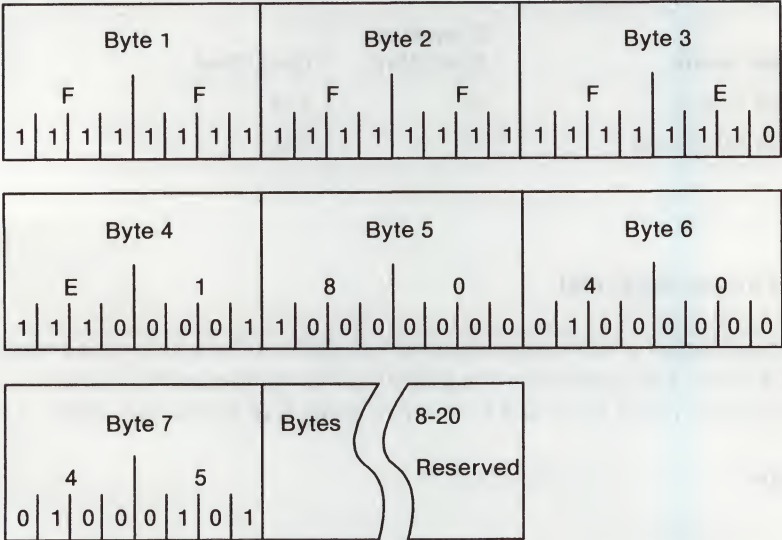
Format

Byte 1 Graphic Order								Byte 2 Graphic Data								Byte 3 Graphic Data							
8				0				x				x				x				x			
1	0	0	0	0	0	0	0	0	1	a	a	a	a	a	a	0	1	a	a	a	a	a	a

RB3F014

Byte	Field Content	Field Description
1	Hex 80	Order code.
2, 3	Graphics data	The last 6 bits of the 2 data bytes form an offset between 0 and 1919 (0 corresponds to row 1, column 1); where the status information is written in the alphanumeric display buffer.

The 20 bytes of read status data have the following format:



RB3F015

Byte	Status Data Description
1, 2	2-byte error code of the last graphics error detected. These codes are described in the <i>User's Guide</i> .
3	Model Identification: <div> <div>Hex Value</div> <div>Model</div> <div>FF5292-2 or Emulation program</div> </div>

Byte	Status Data Description										
4	<p>Graphics Level Number:</p> <table> <tr> <th>Hex Value</th><th>Description</th></tr> <tr> <td>F1</td><td>5292-2 Level 1</td></tr> <tr> <td>F2</td><td>5292-2 Level 2</td></tr> <tr> <td>E1</td><td>Emulation Level 1</td></tr> <tr> <td>E2</td><td>Emulation Level 2</td></tr> </table>	Hex Value	Description	F1	5292-2 Level 1	F2	5292-2 Level 2	E1	Emulation Level 1	E2	Emulation Level 2
Hex Value	Description										
F1	5292-2 Level 1										
F2	5292-2 Level 2										
E1	Emulation Level 1										
E2	Emulation Level 2										
5	<p>Attached Printer Type:</p> <table> <tr> <th>Hex Value</th><th>Description</th></tr> <tr> <td>80</td><td>Not able to identify printer. Printer support is not provided.</td></tr> </table>	Hex Value	Description	80	Not able to identify printer. Printer support is not provided.						
Hex Value	Description										
80	Not able to identify printer. Printer support is not provided.										
6, 7	<p>When an error is detected, this 2-byte value represents a 1-byte offset into the graphics buffer to the last byte processed.</p> <table> <tr> <th>Hex Value</th><th>Description</th></tr> <tr> <td>FFFF</td><td>No error pending.</td></tr> <tr> <td>4x4y</td><td>Error pending; error buffer offset of xy (for example, hex 4045 is the sixth byte).</td></tr> </table>	Hex Value	Description	FFFF	No error pending.	4x4y	Error pending; error buffer offset of xy (for example, hex 4045 is the sixth byte).				
Hex Value	Description										
FFFF	No error pending.										
4x4y	Error pending; error buffer offset of xy (for example, hex 4045 is the sixth byte).										
8 to 20	Reserved (hex 40).										

Notes:

1. The complete 20 bytes of status data may not always be written. Data written to the display buffer for a *read status* order terminates when all 20 bytes are written, when the end of the display screen has been reached, or when a field attribute has been reached.
2. All data written to the display buffer as a result of a *read status* order are encoded above hex 3F to prevent possible work station controller problems.
3. All pending errors are cleared when a *read status* order has been received.

Read Attributes Order (81)

The *read attributes* order gathers information about the characteristics of the graphics display and adapter currently used to run Host Graphics Support. The information is stored in the alphanumeric buffer at the offset specified by the 2 data bytes that follow the order.

Format

Byte 1 Graphic Data								Byte 2 Graphic Data								Byte 3 Graphic Data							
8				1				x				x				x				x			
1	0	0	0	0	0	0	1	0	1	a	a	a	a	a	a	0	1	a	a	a	a	a	a

RB4L035

Byte	Field Content	Field Description
1	Hex 81	Order code.
2, 3	Graphics data	The last 6 bits of the 2 data bytes form an offset between 0 and 1919 (0 corresponds to row 1, column 1), where the attribute information is written in the alphanumeric display buffer.

For example, if the horizontal resolution of the graphics display is 640, then the maximum horizontal pixel count is 639 ($639 = 640 - 1$), which is hex 27F. Bytes 1 through 4 of the attribute information would appear as follows:

Byte 1 Graphic Data								Byte 2 Graphic Data							
4				0				4				0			
0	1	0	0	0	0	0	0	0	1	0	0	0	0	0	0

Byte 3 Graphic Data								Byte 4 Graphic Data							
4				6				7				F			
0	1	0	0	0	1	1	0	0	1	1	1	1	1	1	1

RB4L036

Bytes	Attribute Data Description
1 to 4	Maximum horizontal pixel count
5 to 8	Maximum vertical pixel count
9 to 12	Number of screen colors
13 to 16	Color palette size
17 to 18	Number of characters per row
19 to 20	Number of characters per column
21 to 22	Horizontal pixel size in microns
23 to 24	Vertical pixel size in microns
25 to 26	Number of line widths supported
27 to 40	Reserved

Notes:

1. The complete 40 bytes of attribute data may not always be written. Data written to the display buffer for a *read attribute* order terminates when all 40 bytes are written, when the end of the display screen has been reached, or when a field attribute has been reached.
2. All data written to the display buffer as a result of a *read attribute* order are encoded above hex 3F to prevent possible work station controller problems.

Printer Orders

The printer is not supported in graphics mode.

Order Name	Graphics Byte (Hex)	Described
Printer Data Follows	C0	Causes P1 error
Screen Copy	C1	Causes P1 error
Load Printer A/N Color Mix Table	C2	No
Load Printer Graphics Color Mix Table	C3	No
Set Printer Time-out	C4	No

IEEE Interface Orders

Note: The description of the IEEE orders assumes you understand the IEEE standards and the 9914 Adapter.

Order Name	Graphics Byte (Hex)	Described
Set Address	D1	No
Set Talker Time-out	D2	No
Set EOI Mode	D3	No
IEEE Data Follows	E0	Yes
Take Control Asynchronously	E1	Yes
Go To Standby	E2	Yes
Remote Enable	E3	No
Remote Disable	E4	No
Interface Clear	E5	Yes

Figure 5-3. IEEE Local Command Orders

The host data stream is in IEEE-488 format and the plotter is attached via the RS232 interface. The Emulation program handles the conversion.

IEEE Data Follows Order (E0)

The *IEEE data follows* order sends passthrough data directly to an IEEE device. IEEE data is described in the *IBM 5250 Information Display System Functions Reference Manual*. Data following this order is passed through until an *end of data* is encountered. If the data spans blocks, use the *more data to come* order described in the *IBM 5250 Information Display System Functions Reference Manual*.

IEEE data follows order passthrough data following a *take control asynchronously* order and preceding a *go to standby* order is processed as interface messages. All other *IEEE data follows* order passthrough data is processed as device dependent messages and is sent to the IEEE device.

Format

Byte 1 IEEE Order	Byte 2 through n IEEE Data
E 0	4 x
1 1 1 0 0 0 0 0	0 1 0 0 x x x x

RB3F016

Byte	Field Content	Field Description
1	Hex E0	Order code
2, 3	IEEE data	Passthrough data pair 1
4, 5	IEEE data	Passthrough data pair 2
6, n	IEEE data	Additional passthrough data pairs
n + 1	Hex 92	End of data order

Notes:

1. The interface messages (IM) data is the data that configures the listeners and the talkers on the IEEE bus.
2. The device dependent messages (DDM) data is the data associated with a particular plotter.

Example: This example shows a valid IEEE order sequence. IEEE set orders must have been previously received and processed.

E5 E3 E1 E0 (IM) 92 E2 E0 (DDM) 92

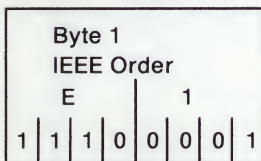
IBM Personal System/2 Mapping: This order is used to process IEEE data transmissions. All interface data is captured and checked for proper IEEE syntax and device configurations. All other data is considered plotter data and is passed directly to the plotter.

Note: If the plotter address contained in the IM data does not match the address specified in the configuration file, an E1 error occurs when attempting to send data to the plotter.

Take Control Asynchronously Order (E1)

The *take control asynchronously* order sends the *take control asynchronously auxiliary* command to the IEEE adapter.

Format



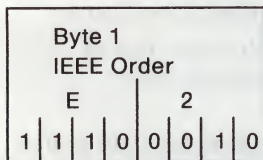
RB3F017

IBM Personal System/2 Mapping: This order defines the start of the IM data stream.

Go To Standby Order (E2)

The *go to standby* order sends the *go to standby auxiliary* command to the IEEE adapter.

Format



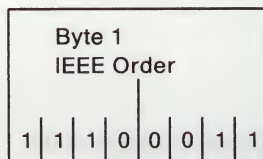
RB3F018

IBM Personal System/2 Mapping: This order marks the end of the IM data stream.

Interface Clear Order (E5)

The *interface clear* order sends an *interface clear auxiliary* command to the IEEE adapter.

Format



RB3F019

IBM Personal System/2 Mapping: This order removes the Emulation program talker from the interface and performs the *take control asynchronously* order function.

Chapter 6. Graphics Display

Introduction 6-2

Graphics Display Differences 6-3

Device Driver Specifications 6-6

Introduction

This chapter gives information about graphics displays.

Notes:

1. Occasionally, a brief flash of unreadable characters may appear on the screen during graphics processing. This is a temporary condition and does not affect operation of either the Emulation program or Host Graphics Support.
2. Under certain circumstances, a polygon may become too complex to fill. At this point, the local error **L3** appears on the status line, the fill mode is set to Hollow, and graphics processing continues.
3. When a display adaptor/monitor combination that supports fewer than eight colors is used, the screen may flash in different colors during graphics processing. This is a result of having too few colors during color mapping, and does not affect operation of either the Emulation program or Host Graphics Support.

Graphics Display Differences

The following table lists the differences in the displays produced by an IBM Personal System/2 running the Emulation program and an IBM 5292 Model 2 (5292-2) Display Station.

Function	IBM 5292-2	Emulation Program
Control of graphics displays	Turned on and off.	May be turned on and off by Local Select mode 1. Simulated by saving and rebuilding the screen. Rebuild time depends on the complexity of the graphics and orders used. This may be 30 seconds or longer. Order stream can overflow the internal buffer causing loss of part of a redisplayed chart.
Graphics coordinates		Scaled to fit current screen width, height, and aspect ratio. Text may be shifted with respect to graphics when alphanumeric data overlays graphics data.
Line style		Provided by VDI and may not match.
Line style mapping		Specified during configuration.
Color mapping		Specified during configuration for displays using fewer than eight colors.

Figure 6-1 (Part 1 of 3). Display Differences

Function	IBM 5292-2	Emulation Program
Polygon fill	<p>Reference line: vertical; follow polygon edge; follow +45° from vertical; follow -45° from vertical.</p> <p>Fill mode: solid boundary and style specified in <i>set style</i> order, solid boundary only, fill with style specified in <i>set style</i> order, boundary only with style specified in <i>set style</i> order.</p>	<p>Reference line: not supported by VDI.</p> <p>Fill mode: solid boundary and style specified in <i>set style</i> order and current line style; solid boundary only; fill with style specified in <i>set style</i> order and current line style; boundary only with style specified in <i>set style</i> order and current line style.</p>
Scanline data		Scaled horizontally and vertically to match current screen width. This may cause bunched or skipped lines.
<i>Set line weight</i> order	Supported by display.	Not supported.
Status line control	Always displayed.	May be turned on and off by Local Select mode 2.
Styled line	Does not write missing or off line segment (area between dashes not painted).	Missing line segments may be painted with background color.
Text attributes and colors		May appear different when in graphics state due to color characteristics of the IBM Personal System/2 hardware.

Figure 6-1 (Part 2 of 3). Display Differences

Function	IBM 5292-2	Emulation Program
Write background	Uses current line style and writing mode.	Clears and then paints screen with solid line style and replace writing mode.
Write polymarker		Always uses replace function.
XOR and OR writing modes		Boolean logic is performed on the color indexes. VDI uses different index values for color selection than the 5292-2. This can result in different output colors.
Blink, underscore, and column separators		Not supported in graphics state.
Vertical scaling factor		Can be altered by Local Select mode 3.
Local Select modes		Supported for Erase Graphics Display and End Graphics processing.

Figure 6-1 (Part 3 of 3). Display Differences

Note: See Chapter 6 in the *User's Guide* for information about Local Select mode.

Device Driver Specifications

The following table lists the specifications for the IBM Personal System/2 device drivers.

VDI Device Driver	# of Colors or Shades	Resolution	CPL
VDIDY011.SYS	2	640 by 480	80
VDIDY012.SYS	16	640 by 480	80
VDIDY013.SYS	16 or 256 (see Note)	320 by 200	40
VDIDYAF1.SYS	16 or 256 (see Note)	640 by 480	80
VDIDYAF2.SYS	16 or 256 (see Note)	1024 by 768	80
Note: The number of colors depends on the type of hardware adapter in the IBM Personal System/2. See your hardware reference for more information.			

Figure 6-2. VDI Files

Appendix A. Character Sets

IBM Personal Computer Printer Character Conversion	A-2
Scan Codes	A-3
IBM Personal System/2 Keyboards and Scan Codes	A-3
IBM 5250 Keyboard and Scan Codes	A-7
Character Set Tables	A-9
EBCDIC Character Set Tables	A-11
EBCDIC Table for U.S.A. English, Upper Case English, and Dutch	A-11
EBCDIC Table for German	A-12
EBCDIC Table for Danish and Norwegian	A-13
EBCDIC Table for Finnish and Swedish	A-14
EBCDIC Table for Italian	A-15
EBCDIC Table for Portuguese	A-16
EBCDIC Table for Spanish and Latin American Spanish	A-17
EBCDIC Table for UK English	A-18
EBCDIC Table for French	A-19
Multi-National EBCDIC Table for Belgian, Swiss, and French Canadian	A-20
ASCII Character Sets	A-21
ASCII Table for Codepage 437	A-21
ASCII Table for Codepage 850	A-22
ASCII Table for Codepage 860	A-23
ASCII Table for Codepage 863	A-24
EBCDIC-to-ASCII Conversion Tables	A-25

IBM Personal Computer Printer Character Conversion

Some of the characters supported by the IBM 3196 displays are not supported by the IBM Personal Computer Printers. Therefore, those IBM 3196 characters that cannot be printed must be converted. EBCDIC codes above hex 20 are translated according to the EBCDIC-to-ASCII tables in this appendix. Codes below hex 20 are not translated, but sent to the printer as is. See the printer operator's guide to determine the action resulting from these codes.

The characters that can be printed are listed below.

- Standard alphabetic and special characters have an ASCII equivalent and print normally. These include:

a b c d e f g h i j k l m n o p q r s t u v w x y z
A B C D E F G H I J K L M N O P Q R S T U V W X Y Z
0 1 2 3 4 5 6 7 8 9
- = ! @ # \$ % & * () [] _ + ' " ; : / \ ? . , { }
< > (space)

- Text control characters, such as tabs and carriage returns, are printed.
- When a PrtSc key or an emulated Print key is pressed, the Emulation program checks the hex value of all ASCII characters (see "ASCII Character Sets" on page A-21) on the screen. The Emulation program limits the printing of character graphics to those characters that have a value less than 7E. All other characters are printed as blanks.

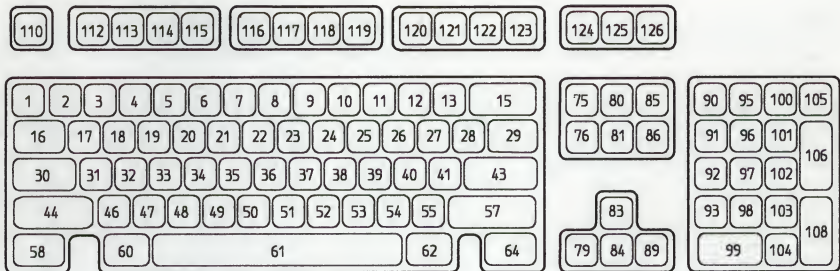
For an attached printer capable of printing characters with a hex value greater than 7E and configured as the attached printer type, the Emulation program automatically changes the limit to allow the additional characters to print.

Scan Codes

A scan code is the number assigned to a key location. The following keyboards and tables show the numbering system for keyboards and the associated hex code for the IBM Personal System/2 and IBM 5250 keyboards.

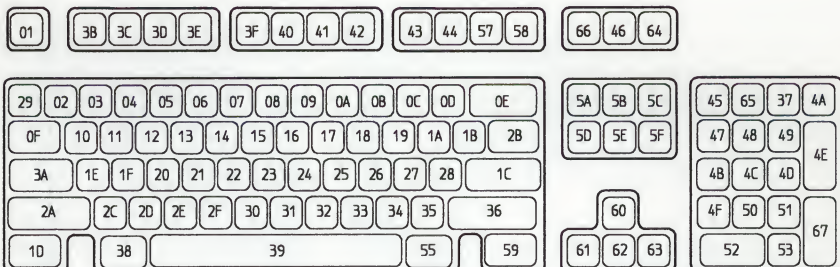
IBM Personal System/2 Keyboards and Scan Codes

The following figures show the keyboard numbering system and scan codes for the U.S.A. (101 keys) and Multi-language (102 keys) versions of the Enhanced keyboard.



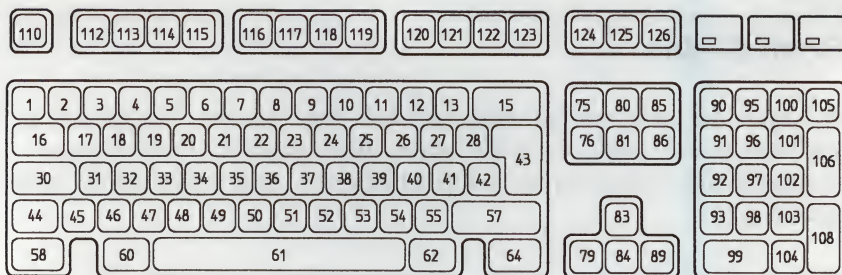
RB4L041

Figure A-1. Key Numbers for the Enhanced Keyboard, U.S.A. Version



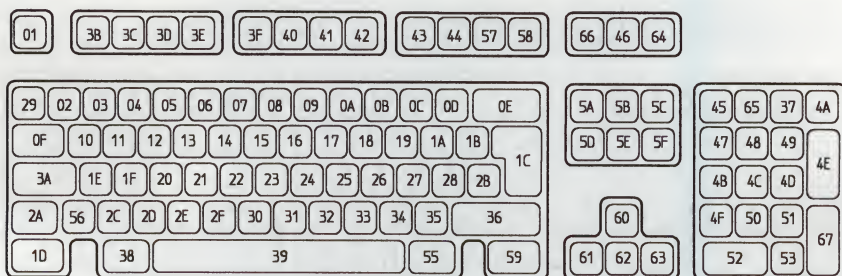
RB4F013

Figure A-2. Scan Codes for the Enhanced Keyboard, U.S.A. Version



RB4L042

Figure A-3. Key Numbers for the Enhanced Keyboard, Multi-Language Version



RB4F012

Figure A-4. Scan Codes for the Enhanced Keyboard, Multi-Language Version

The following tables list the key positions and their corresponding scan codes.

Key Position	PC Scan Code
1	29
2	02
3	03
4	04
5	05
6	06
7	07
8	08
9	09
10	0A
11	0B
12	0C
13	0D
15	0E
16	0F
17	10
18	11

Key Position	PC Scan Code
19	12
20	13
21	14
22	15
23	16
24	17
25	18
19	26
27	1A
28	1B
29	2B
30	3A
31	1E
32	1F
33	20
34	21
35	22

Key Position	PC Scan Code
36	23
37	24
38	25
39	26
40	27
41	28
42	2B *
43	1C
44	2A
45	56 *
46	2C
47	2D
48	2E
49	2F
50	30
51	31
52	32

RB4F003

* This key is present only on the Multi-language keyboard.

Figure A-5. PC Scan Codes, Key Positions 1 through 52

Key Position	PC Scan Code
53	33
54	34
55	35
57	36
58	1D
60	38
61	39
62	55 *
64	59
75	5A *
76	5D *
79	61 *
80	5B *
81	5E *
83	60 *
84	62 *
85	5C *
86	5F *

Key Position	PC Scan Code
89	63
90	45
91	47
92	4B
93	4F
95	65 *
96	48
97	4C
98	50
99	52
100	37
101	49
102	4D
103	51
104	53
105	4A
106	4E

Key Position	PC Scan Code
108	67
110	01
112	3B
113	3C
114	3D
115	3E
116	3F
117	40
118	41
119	42
120	43
121	44
122	57
123	58
124	66 *
125	46
126	64 *

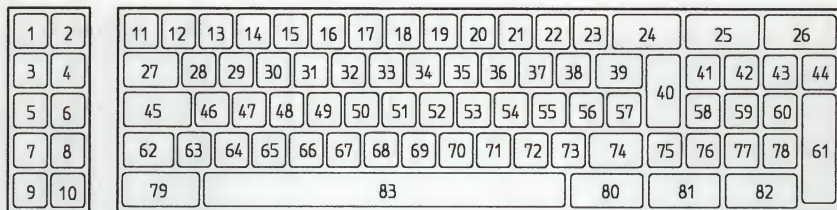
RB4F004

* This scan code is not produced by the physical keyboard.
It is an emulated scan code value.

Figure A-6. PC Scan Codes, Key Positions 53 through 126

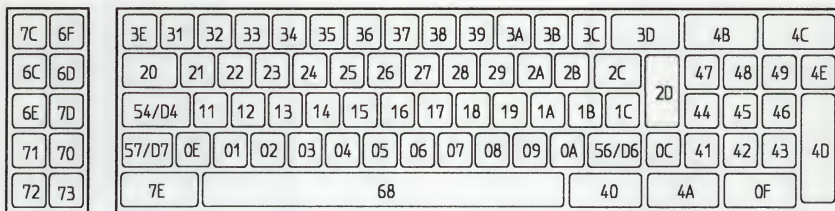
IBM 5250 Keyboard and Scan Codes

The following figures show the keyboard numbering system and scan codes for the IBM 5250 keyboard.



RB4L040

Figure A-7. Key Positions for the IBM 5250 Keyboard



RB4L039

Figure A-8. Scan Codes for the IBM 5250 Keyboard

Note: The Emulation program uses 5250 scan codes rather than 3196 scan codes to maintain compatibility with the Enhanced 5250 Emulation Personal Computer Application Program Interface and the IBM 5292 Model 2 Color Display Station.

The following table lists the key positions and their scan codes.

Key Position	5250 Scan Code
1	7C
2	6F
3	6C
4	6D
5	6E
6	7D
7	71
8	70
9	72
10	73
11	3E
12	31
13	32
14	33
15	34
16	35
17	36
18	37
19	38
20	39
21	3A
22	3B
23	3C
24	3D
25	4B
26	4C
27	20
28	21

Key Position	5250 Scan Code
29	22
30	23
31	24
32	25
33	26
34	27
35	28
36	29
37	2A
38	2B
39	2C
40	2D
41	47
42	48
43	49
44	4E
45 *	54/d4
46	11
47	12
48	13
49	14
50	15
51	16
52	17
53	18
54	19
55	1A
56	1B

Key Position	5250 Scan Code
57	1C
58	44
59	45
60	46
61	4D
62 *	57/d7
63	0E
64	01
65	02
66	03
67	04
68	05
69	06
70	07
71	08
72	09
73	0A
74 *	56/d6
75	0C
76	41
77	42
78	43
79	7E
80	68
81	40
82	4A
83	0F

RB4F005

* Indicates a make/break key. When a make/break key is held down, the first scan code occurs. When the key is released, the second scan code occurs.

Figure A-9. 5250 Scan Codes

Character Set Tables

This section describes the host EBCDIC and IBM Personal System/2 ASCII character sets supported by the Emulation program.

Conversion tables translate EBCDIC characters to ASCII characters. The Emulation program provides tables for NLV EBCDIC or MNCS EBCDIC to ASCII 437 or ASCII 850 conversion. "EBCDIC-to-ASCII Conversion Tables" on page A-25 explains how to view and print the active table.

You can change the codepage and create or modify the EBCDIC-to-ASCII translation table. See the *User's Guide* for instructions.

The following table lists the EBCDIC and ASCII code pages in this appendix supported for national languages.

Language Supported	EBCDIC Codepage	Located	ASCII Codepage	Located
U.S.A. English	037	A-11	437	A-21
	037	A-11	850	A-22
Upper Case English	037	A-11	437	A-21
	037	A-11	850	A-22
UK English	285	A-18	437	A-21
	285	A-18	850	A-22
Belgian Dutch MNCS	500	A-20	437	A-21
	500	A-20	850	A-22
Belgian French MNCS	500	A-20	437	A-21
	500	A-20	850	A-22
French Canadian MNCS	500	A-20	437	A-21
	500	A-20	863	A-24
Danish NLV	277	A-13	437 (see Note)	A-21
	277	A-13	850	A-22
Finnish NLV	278	A-14	437	A-21
	278	A-14	850	A-22

Language Supported	EBCDIC Codepage	Located	ASCII Codepage	Located
French NLV	297 297	A-19 A-19	437 850	A-21 A-22
German NLV	273 273	A-12 A-12	437 850	A-21 A-22
Italian NLV	280 280	A-15 A-15	437 850	A-21 A-22
Dutch NLV (Netherlands)	037 037	A-11 A-11	437 850	A-21 A-22
Norwegian NLV	277 277	A-13 A-13	437 (see Note) 850	A-21 A-22
Portuguese	282 282	A-16 A-16	860 850	A-23 A-22
Spanish NLV	284 284	A-17 A-17	437 850	A-21 A-22
Latin American Spanish NLV	284 284	A-17 A-17	437 850	A-21 A-22
Swedish NLV	278 278	A-14 A-14	437 850	A-21 A-22
Swiss French MNCS	500 500	A-20 A-20	437 850	A-21 A-22
Swiss German MNCS	500 500	A-20 A-20	437 850	A-21 A-22
Swiss Italian MNCS	500 500	A-20 A-20	437 850	A-21 A-22
Note: This codepage uses Group 2 ROM characters.				

EBCDIC Character Set Tables

EBCDIC Table for U.S.A. English, Upper Case English, and Dutch

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	ø	Ø	°	μ	^	{	}	\	0
	1					Required Space	é	/	É	a	j	~	£	A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					ä	ë	Ä	Ë	c	l	t	·	C	L	T	3
	4					à	è	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v	§	E	N	V	5
	6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
	7					å	ï	Å	Ï	g	p	x	¼	G	P	X	7
	8					ç	ì	Ç	Ì	h	q	y	½	H	Q	Y	8
	9					ñ	ß	Ñ	`	l	r	z	¾	I	R	Z	9
	A					¢	!	¡	:	«	ä	í	[-	1	2	3
	B					.	\$,	#	»	ó	¿]	ô	û	Ô	Û
	C					<	*	%	@	ð	æ	Ð	-	ö	ü	Ö	Ü
	D					()	-	'	ý	þ	Ý	..	ò	ù	Ò	Ù
	E					+	;	>	=	þ	Æ	Þ	'	ó	ú	Ó	Ú
	F						¬	?	"	±		®	=	õ	ÿ	Õ	

RB4F014

Figure A-10. EBCDIC Codepage 00037, Version 0

EBCDIC Table for German

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	ø	Ø	°	μ	¢	ä	ü	Ö	0
	1					Required Space	é	/	É	a	j	ß	£	A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					{	ë	l	Ë	c	l	t	·	C	L	T	3
	4					à	è	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v	@	E	N	V	5
	6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
	7					å	ï	Å	Ï	g	p	x	¼	G	P	X	7
	8					ç	ì	Ç	Ì	h	q	y	½	H	Q	Y	8
	9					ñ	~	Ñ	`	l	r	z	¾	I	R	Z	9
	A					Ä	Ü	ö	:	«	a	i	¬	-	1	2	3
	B					.	\$,	#	»	o	ı		ô	û	Ô	Û
	C					<	*	%	§	ö	æ	Ð	-	ı	}	\]
	D					()	-	'	ý	ı	Ý	¨	ò	ù	Ò	Ù
	E					+	;	>	=	þ	Æ	Þ	'	ó	ú	Ó	Ú
	F					!	^	?	"	±		®	=	õ	ÿ	Õ	

RB4F015

Figure A-11. EBCDIC Codepage 00273, Version 0

EBCDIC Table for Danish and Norwegian

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	!	@	°	μ	¢	æ	å	\	0
	1					Required Space	é	/	É	a	j	ü	£	A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					ä	ë	Ä	Ë	c	l	t	·	C	L	T	3
	4					à	è	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v	§	E	N	V	5
	6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
	7					}	ï	\$	Ï	g	p	x	¼	G	P	X	7
	8					ç	ì	Ç	Ì	h	q	y	½	H	Q	Y	8
	9					ñ	ß	Ñ	`	ı	r	z	¾	I	R	Z	9
	A					#		ø	:	«	<u>a</u>	ı	¬	-	ı	2	3
	B					.	Å	,	Æ	»	<u>o</u>	ı		ô	û	Ô	Û
	C					<	*	%	Ø	ð	{	Ð	-	ö	~	Ö	Ü
	D					()	-	'	ý	ı	Ý	¨	ò	ù	Ò	Ù
	E					+	;	>	=	þ	[Þ	'	ó	ú	Ó	Ú
	F						^	?	"	±]	®	=	õ	ÿ	Õ	

RB4F016

Figure A-12. EBCDIC Codepage 00277, Version 0

EBCDIC Table for Finnish and Swedish

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	ø	Ø	°	μ	¢	ä	å	É	0
	1					Required Space	'	/	\	a	j	ü	£	A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					{	ë	#	Ë	c	l	t	·	C	L	T	3
	4					à	è	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v	[E	N	V	5
	6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
	7					}	ï	\$	Ï	g	p	x	¼	G	P	X	7
	8					ç	ì	Ç	Ì	h	q	y	½	H	Q	Y	8
	9					ñ	ß	Ñ	é	l	r	z	¾	I	R	Z	9
	A					§		ö	:	«	ä	ï	¬	-	1	2	3
	B					·	Å	,	Ä	»	ó	í		ô	û	Ô	Û
	C					<	*	%	Ö	ð	æ	Ð	-	!	~	@	Ù
	D					()	-	'	ý	þ	Ý	¨	ò	ù	Ò	Ù
	E					+	;	>	=	þ	Æ	Þ	'	ó	ú	Ó	Ú
	F					!	^	?	"	±]	®	=	õ	ÿ	Õ	

RB4F017

Figure A-13. EBCDIC Codepage 00278, Version 0

EBCDIC Table for Italian

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	ø	Ø	[μ	¢	à	è	ç	0
	1					Required Space]	/	É	a	j	ì	#	A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					ä	ë	Ä	Ë	c	l	t	•	C	L	T	3
	4					{	}	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v	@	E	N	V	5
	6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
	7					å	ï	Å	Ï	g	p	x	¼	G	P	X	7
	8					\	~	Ç	Ì	h	q	y	½	H	Q	Y	8
	9					ñ	ß	Ñ	ù	l	r	z	¾	I	R	Z	9
	A					°	é	ò	:	«	<u>a</u>	ì	¬	-	1	2	3
	B					.	\$,	£	»	<u>o</u>	¿		ô	û	Ô	Û
	C					<	*	%	§	ð	æ	Ð	-	ö	ü	Ö	Ü
	D					()	-	'	ý	¸	Ý	¨	ı	`	Ò	Ù
	E					+	;	>	=	þ	Æ	Þ	'	ó	ú	Ó	Ú
	F					!	^	?	"	±		®	=	õ	ÿ	Õ	

RB4F018

Figure A-14. EBCDIC Codepage 00280, Version 0

EBCDIC Table for Portuguese

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	ø	Ø	°	μ	¢	ã	'	Ç	0
	1					Required Space	é	/	Ê	a	j	ç	£	A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					ä	ë	Ä	Ë	c	l	t	·	C	L	T	3
	4					à	è	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v	§	E	N	V	5
	6					{	î	#	Î	f	o	w	¶	F	O	W	6
	7					å	ï	Å	Ï	g	p	x	¼	G	P	X	7
	8					~	ì	\	Ì	h	q	y	½	H	Q	Y	8
	9					ñ	ß	Ñ	`	l	r	z	¾	I	R	Z	9
	A					[]	õ	:	«	<u>a</u>	i	¬	-	1	2	3
	B					.	\$,	Ã	»	<u>o</u>	¿		ô	û	Ô	Û
	C					<	*	%	Õ	ð	æ	Ð	-	ö	ü	Ö	Ü
	D					()	-	'	ý	¸	Ý	¨	ò	ù	Ò	Ù
	E					+	;	>	=	þ	Æ	Þ	}	ó	ú	Ó	Ú
	F					!	^	?	"	±		®	=	¡	ÿ	@	

RB4F019

Figure A-15. EBCDIC Codepage 00282, Version 0

EBCDIC Table for Spanish and Latin American Spanish

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	ø	Ø	°	μ	¢	{	}	\	0
	1					Required Space	é	/	É	a	j	¨	£	A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					ä	ë	Ä	Ë	c	l	t	•	C	L	T	3
	4					à	è	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v	§	E	N	V	5
	6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
	7					å	ï	Å	Ï	g	p	x	¼	G	P	X	7
	8					ç	ì	Ç	Ì	h	q	y	½	H	Q	Y	8
	9					ı	ß	#	`	ı	r	z	¾	I	R	Z	9
	A					[]	ñ	:	«		ı	^	-	ı	2	3
	B					.	\$,	Ñ	»		ı	!	ô	û	Ô	Û
	C					<	*	%	@	ð	æ	Ð	-	ö	ü	Ö	Ü
	D					()	—	'	ý	ı	Ý	~	ò	ù	Ò	Ù
	E					+	;	>	=	þ	Æ	Þ	'	ó	ú	Ó	Ú
	F						¬	?	"	±		®	=	õ	ÿ	Õ	

RB4F020

Figure A-16. EBCDIC Codepage 00284, Version 0

EBCDIC Table for UK English

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	ø	Ø	°	μ	¢	{	}	\	0
	1					Required Space	é	/	É	a	j	-	[A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					ä	ë	Ä	Ë	c	l	t	•	C	L	T	3
	4					à	è	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v	§	E	N	V	5
	6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
	7					å	ï	Å	Ï	g	p	x	¼	G	P	X	7
	8					ç	ì	Ç	Ì	h	q	y	½	H	Q	Y	8
	9					ñ	ë	Ñ	`	l	r	z	¾	I	R	Z	9
	A					\$!	!	:	«	<u>a</u>	i	^	-	1	2	3
	B					.	£	,	#	»	<u>o</u>	¿]	ô	û	Ô	Û
	C					<	*	%	@	ð	æ	Ð	~	ö	ü	Ö	Ü
	D					()	-	'	ý	¸	Ý	¨	ò	ù	Ò	Ù
	E					+	;	>	=	þ	Æ	Þ	'	ó	ú	Ó	Ú
	F						¬	?	"	±		®	=	õ	ÿ	Õ	

RB4F021

Figure A-17. EBCDIC Codepage 00285, Version 0

EBCDIC Table for French

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	ø	Ø	['	¢	é	è	ç	0
	1					Required Space	{	/	Ê	a	j	..	#	A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					ä	ë	Ä	Ë	c	l	t	•	C	L	T	3
	4					@	}	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v]	E	N	V	5
	6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
	7					å	ï	Å	Ï	g	p	x	¼	G	P	X	7
	8					\	ì	Ç	Ì	h	q	y	½	H	Q	Y	8
	9					ñ	ß	Ñ	µ	l	r	z	¾	I	R	Z	9
	A					°	§	ù	:	«	<u>a</u>	ì	¬	-	1	2	3
	B					.	\$,	£	»	<u>o</u>	¿		ô	û	Ô	Û
	C					<	*	%	à	ð	æ	Ð	-	ö	ü	Ö	Ü
	D					()	-	'	ý	¸	Ý	~	ò	í	Ò	Ù	
	E					+	;	>	=	þ	Æ	Þ	'	ó	ú	Ó	Ú
	F					!	^	?	"	±		®	=	õ	ÿ	Õ	

RB4F022

Figure A-18. EBCDIC Codepage 00297, Version 0

Multi-National EBCDIC Table for Belgian, Swiss, and French Canadian

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0					Space	&	-	ø	Ø	°	μ	¢	{	}	\	0
	1					Required Space	é	/	É	a	j	~	£	A	J	Num Space	1
	2					â	ê	Â	Ê	b	k	s	¥	B	K	S	2
	3					ä	ë	Ä	Ë	c	l	t	·	C	L	T	3
	4					à	è	À	È	d	m	u	f	D	M	U	4
	5					á	í	Á	Í	e	n	v	§	E	N	V	5
	6					ã	î	Ã	Î	f	o	w	¶	F	O	W	6
	7					å	ï	Å	Ï	g	p	x	¼	G	P	X	7
	8					ç	ì	Ç	Ì	h	q	y	½	H	Q	Y	8
	9					ñ	ß	Ñ	`	l	r	z	¾	I	R	Z	9
	A					[]	!	:	«	ä	ï	¬	-	1	2	3
	B					.	\$,	#	»	ó	¿		ô	û	Ô	Û
	C					<	*	%	@	ð	æ	Ð	-	ö	ü	Ö	Ü
	D					()	-	'	ý	¸	Ý	¨	ò	ù	Ò	Ù
	E					+	;	>	=	þ	Æ	Þ	'	ó	ú	Ó	Ú
	F					!	^	?	"	±		®	=	õ	ÿ	Õ	

RB4F023

Figure A-19. EBCDIC Codepage 00500, Version 0

ASCII Character Sets

The Emulation program provides support for one of the following ASCII character set tables, depending on the national language supported:

Codepage	Support
437	Default codepage
850	IBM Personal System/2 multilingual codepage
860	Portuguese codepage
863	French Canadian codepage

ASCII Table for Codepage 437

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0	BLANK (NULL)	►	BLANK (SPACE)	0	@	P	'	p	Ç	É	á				α	≡
	1	☺	◄	!	1	A	Q	a	q	ü	æ	í				β	±
	2	☹	↑	"	2	B	R	b	r	é	Æ	ó				Γ	≥
	3	♥	!!	#	3	C	S	c	s	â	ô	ú				π	≤
	4	♦	¶	\$	4	D	T	d	t	ä	ö	ñ				Σ	∫
	5	♣	§	%	5	E	U	e	u	à	ò	Ñ				σ	∫
	6	♠	■	&	6	F	V	f	v	å	û	ä				μ	÷
	7	•	↓	'	7	G	W	g	w	ç	ù	ö				τ	≈
	8	◼	↑	(8	H	X	h	x	ê	ÿ	ï				Φ	°
	9	○	↓)	9	I	Y	i	y	ë	Ö	Γ				Θ	•
	A	◉	→	*	:	J	Z	j	z	è	Ü	┐				Ω	•
	B	♂	←	+	;	K	[k	{	ï	ø	½				δ	√
	C	♀	└	,	<	L	\	l		î	£	¼				∞	n
	D	♪	↔	-	=	M]	m	}	ì	Ÿ	ì				φ	2
	E	♫	▲	.	>	N	^	n	~	Ä	Pts	«				€	■
	F	☼	▼	/	?	O	_	o	□	Å	f	»				∩	BLANK 'FF'

RB4F024

Figure A-20. ASCII Codepage 437

ASCII Table for Codepage 850

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0	BLANK (NULL)	▶	BLANK (SPACE)	0	@	P	'	p	Ç	É	á			ð	Ó	—
	1	☺	◀	!	1	A	Q	a	q	ü	æ	í			Ð	ß	±
	2	☹	↕	"	2	B	R	b	r	é	Æ	ó			Ê	Ô	=
	3	♥	!!	#	3	C	S	c	s	â	ô	ú			Ë	Ò	¾
	4	♦	¶	\$	4	D	T	d	t	ä	ö	ñ			È	õ	¶
	5	♣	§	%	5	E	U	e	u	à	ò	Ñ	Á		ı	Õ	§
	6	♠	■	&	6	F	V	f	v	å	û	ä	Â	ã	Í	μ	÷
	7	•	↕	'	7	G	W	g	w	ç	ù	ó	À	Ã	Î	þ	,
	8	●	↑	(8	H	X	h	x	ê	ÿ	ı	©		İ	þ	°
	9	○	↓)	9	I	Y	i	y	ë	Ö	®				Û	¨
	A	◉	→	*	:	J	Z	j	z	è	Ü	¬				Û	·
	B	♂	←	+	;	K	[k	{	ï	ø	½				Ù	1
	C	♀	└	,	<	L	\	l		î	£	¼				ý	3
	D	♪	↔	-	=	M]	m	}	ì	Ø	ı	¢		ı	Ý	2
	E	🎵	▲	.	>	N	^	n	~	Ä	x	«	¥		İ	—	■
	F	☼	▼	/	?	O	_	o	□	Å	f	»		☒		/	BLANK 'FF'

RB4F025

Figure A-21. ASCII Codepage 850

ASCII Table for Codepage 860

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0	BLANK (NULL)	►	BLANK (SPACE)	0	@	P	‘	p	Ç	É	á				α	≡
	1	☺	◄	!	1	A	Q	a	q	ü	À	í				β	±
	2	☹	↕		2	B	R	b	r	é	È	ó				Γ	≥
	3	♥	!!	#	3	C	S	c	s	â	ô	ú				π	≤
	4	♦	¶	\$	4	D	T	d	t	ã	õ	ñ				Σ	∫
	5	♣	§	%	5	E	U	e	u	à	ò	Ñ				σ	∫
	6	♠	■	&	6	F	V	f	v	Á	Ú	ä				μ	÷
	7	•	↓	'	7	G	W	g	w	ç	ù	ü				τ	≈
	8	◼	↑	(8	H	X	h	x	ê	ì	í				Φ	°
	9	○	↓)	9	I	Y	i	y	Ê	Õ	Ö				Θ	•
	A	◉	→	*	:	J	Z	j	z	è	Ü	ÿ				Ω	•
	B	♂	←	+	;	K	[k	{	Í	℄	½				δ	✓
	C	♀	└	,	<	L	\	l		Ô	£	¼				∞	n
	D	♪	↔	-	=	M]	m	}	ì	Ù	í				φ	₂
	E	🎵	▲	.	>	N	^	n	~	Ã	Pts	«				€	■
	F	☼	▼	/	?	O	_	o	◊	Â	Ó	»				∩	

RB4F026

Figure A-22. ASCII Codepage 860

ASCII Table for Codepage 863

		First Hexadecimal Character															
		0	1	2	3	4	5	6	7	8	9	A	B	C	D	E	F
Second Hexadecimal Character	0	BLANK (NULL)	►	BLANK (SPACE)	0	@	P	'	p	Ç	É	!				α	≡
	1	☺	◄	!	1	A	Q	a	q	ü	È	'				β	±
	2	☹	↕	!!	2	B	R	b	r	é	Ê	ó				Γ	≥
	3	♥	!!	#	3	C	S	c	s	â	ô	ú				π	≤
	4	♦	¶	\$	4	D	T	d	t	Â	Ë	..				Σ	f
	5	♣	§	%	5	E	U	e	u	à	Ï	,				σ	∫
	6	♠	=	&	6	F	V	f	v	¶	û	3				μ	÷
	7	•	↕	'	7	G	W	g	w	ç	ù	-				τ	≈
	8	•	↑	(8	H	X	h	x	ê		Î				Φ	°
	9	○	↓)	9	I	Y	i	y	ë	Ô	Γ				Θ	•
	A	○	→	*	:	J	Z	j	z	è	Ü	Γ				Ω	•
	B	♂	←	+	;	K	[k	{	ï	ø	½				δ	√
	C	♀	└	,	<	L	\	l		î	£	¼				∞	n
	D	♪	↔	-	=	M]	m	}	=	Ù	¾				φ	2
	E	♪	▲	.	>	N	^	n	~	À	Û	«				€	■
	F	☼	▼	/	?	O	_	o	□	§	f	»				∩	BLANK 'FF'

RB4F027

Figure A-23. ASCII Codepage 863

EBCDIC-to-ASCII Conversion Tables

To view the conversion table for EBCDIC to ASCII, execute this procedure:

1. Access the configuration program.
2. Select the advanced option, *Modify EBCDIC-to-ASCII Translation Table*. See Chapters 3 and 11 in the *User's Guide* for information about the configuration program and this option.
3. If you need a hard copy of the table, print the screen.

UNIT 1: THE HISTORY OF THE UNITED STATES

The following table lists the major events in the history of the United States from 1776 to 1865.

Year	Event
1776	Declaration of Independence
1787	Constitution of the United States
1791	Bill of Rights
1800	Jefferson's Purchase
1803	Louisiana Purchase
1812	War of 1812
1820	Missouri Compromise
1845	Texas Annexation
1848	Texas Independence
1850	Compromise of 1850
1861	Secession of Southern States
1862	Emancipation Proclamation
1865	End of the Civil War

Appendix B. Decimal to Hexadecimal Conversion

This table aids in converting decimal values to hexadecimal values.

		First Decimal Digit(s)												
Second Decimal Digit		0	1	2	3	4	5	6	7	8	9	10	11	12
	0	0	A	14	1E	28	32	3C	46	50	5A	64	6E	78
	1	1	B	15	1F	29	33	3D	47	51	5B	65	6F	79
	2	2	C	16	20	2A	34	3E	48	52	5C	66	70	7A
	3	3	D	17	21	2B	35	3F	49	53	5D	67	71	7B
	4	4	E	18	22	2C	36	40	4A	54	5E	68	72	7C
	5	5	F	19	23	2D	37	41	4B	55	5F	69	73	7D
	6	6	10	1A	24	2E	38	42	4C	56	60	6A	74	7E
	7	7	11	1B	25	2F	39	43	4D	57	61	6B	75	7F
	8	8	12	1C	26	30	3A	44	4E	58	62	6C	76	80
	9	9	13	1D	27	31	3B	45	4F	59	63	6D	77	81

RB3F020

Figure B-1 (Part 1 of 2). Decimal to Hexadecimal Conversion

		First Decimal Digit(s)												
Second Decimal Digit		13	14	15	16	17	18	19	20	21	22	23	24	25
	0	82	8C	96	A0	AA	B4	BE	C8	D2	DC	E6	F0	FA
	1	83	8D	97	A1	AB	B5	BF	C9	D3	DD	E7	F1	FB
	2	84	8E	98	A2	AC	B6	C0	CA	D4	DE	E8	F2	FC
	3	85	8F	99	A3	AD	B7	C1	CB	D5	DF	E9	F3	FD
	4	86	90	9A	A4	AE	B8	C2	CC	D6	E0	EA	F4	FE
	5	87	91	9B	A5	AF	B9	C3	CD	D7	E1	EB	F5	FF
	6	88	92	9C	A6	B0	BA	C4	CE	D8	E2	EC	F6	
	7	89	93	9D	A7	B1	BB	C5	CF	D9	E3	ED	F7	
	8	8A	94	9E	A8	B2	BC	C6	D0	DA	E4	EE	F8	
	9	8B	95	9F	A9	B3	BD	C7	D1	DB	E5	EF	F9	

RB3F021

Figure B-1 (Part 2 of 2). Decimal to Hexadecimal Conversion

For example: decimal **132** equals hex **84**.

Appendix C. Updating the Emulation Program

Introduction	C-2
The Control Data Set	C-2
Updating with the Update Program	C-5
Listing Applied Updates	C-6

Introduction

If you experience problems, IBM may instruct you to update the Emulation program. This appendix explains the commands used to update the program, and how to run the Update program.

Note: IBM will supply the update information. The customer will only need to create or edit a file to include this information, then update the Emulation program.

Updating involves two basic steps:

1. Creating or modifying a file with the update information supplied by IBM. This file is called the control data set.
2. Using the Update program to update the Emulation program with the control data set.

The Control Data Set

A control data set is a file of special commands which give directions to the Update program. (Note that the Update program will allow you to update only one file at a time.) The file may be given any unique name and may be created with any DOS editor that terminates an input line with CR + LF and does not use tabs. Each input line starts a new command for the Update program and has a maximum length of 256 characters. The Update program executes the commands from the control file in order.

All commands in the control data set have positional parameters associated with them, which are separated by blanks. You can enter the commands in upper or lower case.

Following is a list of valid update commands and their parameters:

- B** This command resets any error condition found to this point of execution.

When the apply program detects a mis-compare during a verify command, all following commands are ignored until a new **B** command is executed. Therefore, this command may be used for conditional updating. (Note that nesting is not allowed.)

- C** This command allows you to place comment lines within the control file. Any new line in the control file that begins with **C** will be ignored. For example:

```
C This is a comment line.
```

```
C A comment line is ignored by the program.
```

- P** This command updates part of a file. The first parameter for this command is the hex address to start the verify relative to the beginning of the file. This address may have simple hex addition and subtraction within it. Hex values are right-justified.

The second parameter is the data to update with. If this parameter starts with a single quote, the rest of the parameter is considered a character string and is terminated by another single quote. Otherwise, the parameter is assumed to be hex data in full byte format (2 characters per byte required). If this parameter is in hex format, you may use commas on any byte boundary to separate input data. For example:

```
P 0 'This is the beginning of the file'
```

```
P EE-1 90,90,909090
```

- S** This command places a stamp or ID at the end of a COM or EXE file. It is designed to help track the updates and sequence in which they were applied for a particular program. The only parameter is 5 bytes of ID for the update. This update ID is appended to the end of the COM or EXE file with other control information when executed. (Note that the Update program will enforce the rule of only appending a stamp to COM or EXE files.) For example:

```
S WSE01
```

```
S PAT1Z
```

V This command verifies part of the file that is to be updated. Its first parameter is the hex address to start the verify relative to the beginning of the file. This address may have simple hex addition and subtraction within it. Hex values are right-justified. This feature is useful for handling relative addresses within a file.

The second parameter is the data used to verify. If this parameter starts with a single quote, the rest of the data is considered a character string and is terminated by another single quote. Otherwise, the parameter is assumed to be hex data in full byte format (two characters per byte required). If the parameter is in hex format, you may use commas on any byte boundary to separate input data. For example:

```
V 10+2A-C 'DUMMY STRING'
V 100+1A CF,FA,534678,90,90,DC5C
```

The following is an example of a control data set:

```
C THIS IS THE BEGINNING OF THE
C UPDATE CONTROL DATA SET EXAMPLE
B
V 0 '123'
C IF THE VERIFY AT ADDRESS 0 FAILS
C THEN THE FOLLOWING UPDATE AND STAMP
C WILL NOT OCCUR
P 0 '234'
S 10001

B
V 0 '234'
C IF THE VERIFY AT ADDRESS 0 PASSES
C THEN THE FOLLOWING UPDATE AND STAMP
C WILL OCCUR
P 0 '123'
S 10002
C END OF THE EXAMPLE
```

Updating with the Update Program

Use the following procedure to update the Emulation program.

1. You can run the Update program from the DOS prompt or from the WSE main menu (see Chapter 4 in the *User's Guide*).
 - Change to the emulation directory, if necessary. At the DOS prompt, type WSEPATCH and press Enter.
 - Or, select the option, *Update the Emulation program* from the WSE main menu.

The WSE Update Utility menu appears.

2. Select option 1, *Apply an Update to an WSE File* to update the Emulation program. (Option 2 is explained below.)

The WSE Update application screen appears. This screen prompts you to enter the file to be updated and the file with the update information.

3. Type WSE.COM and press Tab.
4. Enter the file name of the control data set and press Enter.

As the Update program applies updates to the program, a list of update stamps appears on the screen. The reverse image status line at the bottom of the screen will indicate any error condition or completion of the update process.

Listing Applied Updates

The Update program also allows you to view a list of all updates that have been made to a file.

1. Select option 2 from the WSE Update Utility screen. The program prompts you to enter the WSE file name.
2. Type `WSE.COM` and press Enter. A list of all updates applied to the file appears in this format:

```
FIXssssssyyyymmdd
```

where *sssss* is the stamp specified by an **S** command, and *yyyy mm dd* is the date of the update.

Glossary

access. A way or means of approach to a part or to data.

acronym. A word formed from the initial letter or letters of each of the successive parts or major parts of a compound term.

adapter. The part of a device or feature required to attach it to a processing unit.

address. The number assigned to a device or to a location in memory.

alarm. An audible tone at a display or printer that is used to get the operator's attention.

align. To bring into or be in line with another or with others. For example, to line up the numbers on the decimal point.

allocate. To assign a resource, such as a disk file or a diskette file, to perform a specific task.

alphabetic. Of, relating to, or employing an alphabet.

alphabetic keys. The letter keys a through z, blank, comma, period, and hyphen.

alphanumeric. (1) Consisting of both letters and numbers and often other symbols (such as punctuation marks and mathematic symbols).
(2) Capable of using alphanumeric characters.

Alt state. The definition for a key when you press and hold down the Alt key, then press that key.

alternate. Another choice or option.

alternate shift mode. A method of operation that causes keys to be entered as characters or functions (For example, the Backspace key is represented as the clear function when the Alt key is held down). See also *nonshifted mode* and *shifted mode*.

application program. A program used to perform a particular data processing task such as payroll.

arctic. Referring to work stations attached to the host via twinaxial cable using IBM 5250-family protocol.

ASCII. American Standard Code for Information Interchange. The standard code using a coded character set consisting of 7-bit coded characters (8 bits including parity check), used for information interchange among data processing systems, data communications systems, and associated equipment. The ASCII set consists of control characters and graphic characters.

aspect ratio. The ratio of the width of the data to its height as displayed on a monitor or display.

asterisk. The character (*), used as a reference mark in printing or writing.

async. See *asynchronous*.

asynchronous. A method of data communications in which information is sent at irregular intervals. Serial printers use asynchronous communications.

attribute. The properties associated with the use of a device.

authorize. To allow a user to communicate with or make use of an object, resource, or function.

auto-configuration. The automatic arrangement of the parts of a computer.

AUTOEXEC.BAT file. A batch file that contains DOS commands or program names that are executed immediately after DOS is started.

back up. To copy information, usually onto a diskette, for safekeeping.

backspace. To move back a position along a row.

Base state. The definition for the key when you press only that key.

batch file. A file containing one or more commands that DOS executes one at a time. All batch files must have the extension of BAT.

beep. See *alarm*.

binary. Pertaining to a numbering system with a base of 2. Valid digits are 0 (zero) and 1 (one).

binary synchronous. A character-oriented, half-duplex data communications protocol.

bit. The basic unit of computer information, such as 0 or 1.

block cursor. See *cursor*.

boot. Loading and executing the boot record. This is accomplished by pressing the Alternate, Control, and Delete key sequence.

boot record. A record stored on track 0 and sector 0 of a disk(ette) which contains a program designed to load the Disk Operating System into memory and start its execution.

boundary. The division between each of 64K blocks of physical storage addresses.

buffer. A temporary storage unit that accepts data at one rate and delivers it at another rate.

bypass. To avoid by means of another path.

byte. The amount of storage required to represent one character; a byte is 8 bits.

cable splice. The physical connection between two pieces of twinaxial cable.

Cable Thru feature. A special feature that allows more than one display station (except the 5251 Model 2 or 12) and/or more than one printer to be attached to a single cable path.

capacity. The ability to receive, hold, or store.

card. An adapter card that is installed in the IBM Personal System/2 to enable it to be connected to another device or to increase the size of the computer's memory.

caret. The wedge shaped mark that indicates where a character is to be inserted.

CGCS. Coded graphic character set

character codes. The ASCII or EBCDIC values assigned to the symbols or functions that are used by the IBM Personal System/2 or in the IBM 3196 Information Display System.

character set. A set or style of alphabetic, numeric, and special characters that can be displayed or printed by the IBM Personal System/2 or a device in the IBM 3196 Information Display System.

checklist. Any list in which items can be compared, scheduled, verified, or identified.

clicker. A mechanism that sounds when a key is pressed and accepted for processing.

close a file. To remove a file from the program in which it is executing. Contrast with *open a file*.

column. The character position within a print line or on a display. The positions are numbered from 1, by 1 starting at the leftmost

character position and extending to the rightmost position.

column separators. The character symbol displayed at the beginning and end of each column in an input field.

combination. A result of bringing two or more things together.

command line. (1) A DOS command and any associated parameters, file names, and so forth. (2) The line on the display screen following the DOS prompt where you type a DOS command.

compatible. The ability of one device to work with another without the need to change either device.

CONFIG.SYS. A DOS file that contains the device driver assignments.

configuration. The group of machines, devices, and programs that make up a data processing system.

configure. (1) To connect the devices, work stations, programming, and the host system to each other by using addresses. (2) To describe to a system or program the configuration.

contention. The contest of two or more rivals trying to reach the same goal at the same time.

control panel. A panel that allows you to operate a device or a system.

control sequence. A sequence of hex values that instruct the printer

to perform a function or print a particular output.

controller. A device used to coordinate and control the operation of one or more devices.

coordinate. To put things in order.

copyright. The exclusive right to reproduce, publish, and sell the matter and form of a material.

CPI. Characters per inch.

CPL. Characters per line.

current directory. The root directory or the last directory specified by a DOS CHDIR (Change Directory) command.

current screen size. The number of rows and columns of data which are displayed on the screen.

cursor. A movable marker (such as an underline or a block) on the display used to indicate to the operator where to type the next character.

cursor location. The location on the screen (row 1 through 24 and column 1 through 80), at which the cursor is located.

customize. To change to suit personal needs or preferences.

D connector. A D-shaped connector like the connector on the cable that connects the Integrated Cable Assembly to the IBM Personal System/2.

data. All information entered into or used by the computer.

data stream. The sequence of characters being transmitted over a line.

deallocate. To free memory or disk(ette) space from use by a specific task.

default value. A value stored in the system that is used when no other value is specified.

deflector. The person or object that changes the course of travel.

delete. To remove. For example, to erase a file.

determination. The act of making or arriving at a decision.

device. The generic term for a piece of equipment or a mechanism; for example, the IBM Personal System/2 display or the personal computer attached printer.

device address. The unique number assigned to a device that enables it to communicate with other devices in a system.

device driver. A program used to control the operation of an output device.

device type. The number assigned to a unit in a system, such as 5216 (IBM Wheelprinter).

diagnostic. Pertaining to the investigation of the cause of a error.

diagram. A line drawing made to show the location of objects.

digit. Any of the numbers from 0 through 9. Hex digits also include A through F.

directory. A list of files contained on a disk(ette). Directories are used to organize files into related groups.

disable. To prevent a unit of a data processing system from operating with the rest of the system.

disclosure. The act of exposing to view.

disk. A round flat plate coated with a magnetic substance on which data from a computer is stored.

disk drive. See *drive*.

diskette. A thin, flexible magnetic plate that is permanently sealed in a protective cover.

diskette drive. See *drive*.

display screen. The part of a display station on which information is displayed.

display station. A device that includes a keyboard from which an operator can send information to the system and a display screen on which an operator can see the information sent to or the information received from the system. See also *work station*.

DOS. Disk Operating System. A program that works with the processing unit and the disk or diskette drives to control the flow of data.

DOS session. The mode of operation which allows the IBM

Personal System/2 to perform independently of the host system when attached as an emulated 3196 work station.

DP. data processing.

drive. The electromechanical device which records and reads the information that is stored on a disk or diskette.

driver. See *device driver*.

EBCDIC. (1) An acronym for Extended Binary-Coded Decimal Interchange Code. A set of 256 eight-bit characters. (2) A set of standards for data transmission.

emulate. To imitate a device with another device.

emulated control panel. The screen used on the IBM Personal System/2 to imitate the control panel of the IBM 5256 Matrix Printer, 5224 Printer, or 5219 Printer.

emulation. The use of a device or program to perform the functions of another device or program.

emulation session. The mode of operation in which the IBM Personal System/2 is operated like a work station in an IBM 3196 Information Display System.

emulator. A program or device that performs the functions of another program or device.

engineering change. The alteration to a product that allows it to perform either additional functions or to correctly perform its designed functions.

engineering change level. The manufacturing level of a product.

Enhanced keyboard. The 101-key (102-key for the World Trade version) IBM Personal System/2 keyboard.

error code. A hexadecimal value that identifies a condition that interferes with normal operations.

error message. A displayed indication that an error has been detected.

escape sequence. A combination of keystrokes used to send commands to a printer. For example, Esc-C is the ASCII Set Page Length command.

execution. The process of carrying out the instructions of a computer program by a computer.

facility. Something that is built, installed, or established to serve a particular purpose.

FID. Font ID.

field. One or more characters of information (such as a name or an amount).

field exit. To move the cursor from a numeric field on the display screen.

field exit required. A rule that requires either a field exit key, a cursor movement key, or a nondata key be used to move the cursor from a numeric field.

file. A collection of records of information that you create, group, and access by file name.

file name. The name assigned to a file.

fixed address vector. A pointer in the Emulation program that points from fixed addresses to internal addresses that can change.

fixed disk. A disk of rigid material with a magnetic coating used for mass storage of data.

font. An assortment of type all of one size and style.

format. (1) (noun) The specific arrangement of information. (2) (verb) To arrange information in a specific way.

formatted screen. A screen with fields established for specific information.

free key mode. An operating mode that allows the entering of data without control by a system program (unformatted).

function keys. (1) IBM Personal System/2 keys F1 through F12. Their functions are assigned by the operating system or application program. (2) A keyboard key that requests an action but does not display a character. The cursor movement keys are examples of function keys.

function control keys. See *function keys*.

GCGID. Graphic character set global registry ID.

GCID. Graphic character set ID.

generic. Relating to or characteristic of an entire group or class.

GFID. Global font ID.

graphics. See *graphics-capable*.

graphics-capable. The ability to display or print complicated pictures.

half-duplex. A sequential two-way transmission of data. Only one device can transmit at a given time.

hardware. The equipment that makes up a data processing system. Contrast with *software*.

help text. A display aid that provides the operator with an explanation of a message that has been received or that allows an operator to request information on how to use a key, menu, or parameter.

hex. See *hexadecimal*.

hexadecimal. A numbering system with a base of 16. Valid digits range from 0 (zero) through 9 (nine) and A (ten) through F (fifteen).

home position. (1) The first input position of the first input field on the screen of a display station. (2) The position (far left) to which the print head moves after the printer has been turned on, after the Stop switch has been pressed.

host system. (1) The System/36 or System/38. (2) The primary or

controlling system in a data communications configuration.

Hot Key function. A function that allows you to change sessions.

ID. Identification.

IEEE. Institute of Electrical and Electronic Engineers.

II. Input inhibited. The input inhibited indicator used in the Emulation program.

IM. See *insert mode*.

inadvertent. By mistake, unintentional.

incur. Become responsible for.

inhibited. Prevented from doing something.

initialize. To prepare for use. For example, to format a diskette and to initialize registers, and/or program variables to the condition previously determined by a program.

initiate. To cause something to start.

input. Data to be processed.

input field. An area on the screen in which an operator enters data. Input fields are blank on the screen and can have a prompt before the input field.

insert mode. The mode of operation that allows characters to be placed between the characters already displayed on the screen when a character key is pressed.

Characters are inserted at the location identified by the cursor.

installation. The act of installing, the state of being installed.

Integrated Cable Assembly. A cable that connects the adapter to the twinaxial cable. This cable automatically terminates the line when a second line is not connected.

integrity. The state of not being changed or erased.

intensity. The level of brightness used to display the characters on the screen.

interact. To act upon one another.

interface. The machinery and programs that permit the exchange of information between computers or devices.

interrupt. A signal that temporarily stops a process, usually so that another process can be executed.

interrupt level. A path used to get the processing unit's immediate attention.

I/O. Input/output.

I/O address. See *device address*.

justify. To adjust text to be even with the left and right margins.

K. Kilobyte. 1024 bytes of information.

KB. Keystroke buffering. The display indicator used to show the

operator when keystrokes are saved.

keyboard layout. The arrangement of the keys on the keyboard attached to a display station.

keyboard template. See *template*.

keystroke. The act of pressing a key on a keyboard.

keyword. A symbol that identifies a parameter.

KS. Keyboard shift. The display indicator used to show the operator when the keyboard is in the shifted mode.

line. The cable from the host computer or controller to which one or more work stations are attached.

load. To move data or programs into memory.

local display station. See *local work station*.

local work station. A work station that is attached directly to the host system with twinaxial cable.

lowercase mode. The keyboard is in lowercase mode when the Shift keys are not pressed and the Shift Lock, Caps Lock, and Num Lock functions are unlocked.

LPI. Lines per inch.

MB. Megabyte. 1,048,576 bytes of information.

make/break key. A key that sends a scan code to the host system

when it is pressed and again when it is released.

mandatory. Required.

master profile. A file that contains configuration information about all emulation sessions.

matrix. Characters arranged in rows and columns.

maximum. The largest number in a group of numbers.

memory. The devices used to store data within a computer.

minimum. The smallest number in a group of numbers.

mode. A method of operation.

modem. A device that allows information to be exchanged between computers using telephone lines.

monitor. The device that is used to display information that is viewed by a computer operator.

monochrome. Made with a single color.

multiple. More than one.

MW. Message waiting. The display indicator used to show the operator when a message received from the host is waiting to be displayed.

nibble. Half of a byte.

nonshifted mode. A method of operation that causes keys to be entered as characters or functions

similar to when a typewriter's shift key is not held down (for example, the **a** is represented as **a**). See also *alternate mode* and *shifted mode*.

normal image. The display attribute that causes characters to be displayed as light characters on a dark background.

null character. A blank (no character) that is displayed as a result of zeros occupying a character position in the buffer.

numeric. Any of the numbers 0 through 9.

numeric keys. The characters 0 through 9, blank, comma, period, plus, and minus.

on-card memory. The memory contained on the adapter.

offline. The state of not being controlled by, or directly communicating with a host system.

online. The state of being controlled by, or directly communicating with a host system.

open a file. To associate a file with a program. Contrast with *close a file*.

operating system. A program that supervises the execution of user programs by the computer.

overlay. (1) To write over (and therefore destroy) an existing file.
(2) A program segment that is loaded into storage and replaces all or part of a previously loaded program segment.

override. To use in place of another.

panel. A screen display listing options or requesting action.

parallel printer. A printer that uses the parallel printer interface feature.

parameter. A value supplied to a program that is used as input or controls the actions of the program.

password. A code you must enter before you can access specific programs.

path. The sequence of directories designated for DOS to search when looking for a program.

PEL. Picture element.

physical planning. The action taken to prepare a site for the installation of a computer or some related equipment.

pixel. An element of a picture.

playback sequence. A combination of characters and/or functions assigned to a key.

plotter. An output device that graphically records data in two dimensions (length and width) on paper.

port. (1) The part of the system unit or controller to which cables for work stations are attached. (2) The part of the Integrated Cable Assembly to which the work station line may be connected.

print buffer. A storage area within the printer where characters to be printed are temporarily stored.

print head. The part of the printer that strikes the ribbon to print characters on the paper forms.

printer. A device that provides printed output.

printwheel. The element used to create the impressions printed by the IBM 5216 Wheelprinter.

processing unit. The part of a computer that operates on data.

profile. Data file that describes the features of a user, program, or device.

program. A sequence of instructions for a computer.

program diskette. A diskette which contains programs to be executed on the IBM Personal System/2. For example, the Work Station Emulation Program Licensed Program Diskette.

prompt. A displayed request for information or operator action.

protocol. A series of rules for transferring data.

RAM. Random access memory. The volatile, electronic memory in computers in which data is stored temporarily for manipulation.

recovery procedure. The action performed by the operator when an error message appears on the display screen.

regeneration buffer. Data area used to store screen content in EBCDIC format.

remote display station. See *remote work station*.

remote work station. A work station that is attached directly or indirectly to the host system a communication channel.

remotely. Indirectly or from a distance.

reset. See *system reset*.

resolution. The visual quality of the display.

restart. To start again.

retries. To try again.

reverse image. A method of highlighting where the character color on the screen and the background color are swapped. For example, in place of green on black, the characters would appear as black on green.

right adjust. To position a character(s) within a field to the rightmost position of the field.

root directory. The directory on a disk(ette) that contains the list of files stored on that disk(ette). If there is more than one directory on the disk(ette), the root directory can also contain the names of each of the other directories. This directory is established when the disk(ette) is formatted.

row. The horizontal arrangement of characters on a screen.

SA. System available. The system available indicator is used with the Emulation program.

scan code. A numeric representation of a key's location on a keyboard.

screen. A display, similar to a television screen, used to display characters.

SCS. SNA (system network architecture) control string.

self-loading program. A program that does not use DOS to move from a diskette to IBM Personal System/2 memory.

separator. See *column separator*.

serial printer. A printer that uses the serial printer interface feature.

session. (1) The period of time in which the host system and one of the devices are communicating.
(2) The logical connection between the host system and a work station. The session may be established for a display station or for a printer.

session profile. A file which contains configuration information about individual sessions.

set up. To prepare an object to be used.

SGEA. Set graphic error action.

Shift state. The definition for the key when you press that key and a Shift key at the same time. The shift state definition is also used during shift lock, caps lock, or num lock.

shifted mode. A method of operation that causes keys to be entered as a character or function similar to when a typewriter's shift key is held down. (For example, an **a** represented as **A**). See also *alternate mode* and *nonshifted mode*.

sign on. To begin a session at a display station.

sign on. The action an operator uses at a display station in order to begin working at the display station.

sign-on screen. The screen that prompts the user to enter the appropriate sign-on command.

signal. To notify.

signed numeric. A format control used to specify that the units position of the field is reserved for the sign (usually, blank for positive and minus for negative).

signed numeric field. A numeric field that has a character in it to identify it as having either a positive or negative value.

simulate. To imitate.

simultaneous. At the same time.

software. The programs, procedures, and rules that are stored as data but are used to control the operation of a computer. Contrast with *hardware*.

software reset sequence. A keying sequence that allows you to leave emulation mode and take the emulated work station offline.

splice. To join by weaving (as two ropes) or by connecting two objects.

spool. To place in a queue.

stamp. ID.

station address. See *work station address*.

station address list. A list of work station addresses a session may use.

station protector. A device connected to the twinaxial cable to protect a work station against lightning.

status. Condition. For example, the status of a display station, a printer, or a job.

status line. The 25th (or bottom) row on the screen of the IBM Personal System/2, which is used to display the condition of the emulated session.

subsystem. A system that operates under control of another system.

symbol keys. The keys on the keyboard that represent displayable characters other than alphabetic and numeric characters.

synchronous. A type of data communications protocol that transmits data according to a precisely timed clock pulse.

syntax. The rules for the construction of a statement.

system. A computer, its devices, and programs.

system operator. A person who uses a display station, which is designated as the system device to activate certain system functions and control and monitor system operation.

system reset. The reloading of the operating system by pressing the Alternate, Control, and Delete key sequence. This causes the boot record to be read into memory and given control.

system time. The time of day as recognized by a system.

system unit. The part of the IBM Personal System/2 that contains the processing unit.

table. A list of data in which each item can be uniquely identified.

template. A pattern to help the user identify the location of keys on a keyboard.

terminate. Bring to an end.

terminator. A twinax cabling device placed at the end of twinax cabling to prevent line noise.

terminator switch. A switch used to terminate the line at the last work station when the Cable Thru feature is used.

toggle. To operate a device (switch) that stays in the operated position until operated a second time.

trademark. A device (as a word) that identifies an owner of a machine, program, or book.

twinaxial cable. A cable made of two wires inside a shield.

unassigned. Not assigned.

underscore. To draw a line under.

unique. The only one.

update. To modify.

uppercase. Capital letters.

uppercase mode. The method of operation that causes alphabetic characters to be displayed as capital letters and all other keys in the typewriter area, when pressed, to be displayed as the character shown on the upper portion of these keys.

utility. A program in general support of the processes of a computer; for example, a sort program.

variable n. A placeholder for a printing function that can be changed at print time.

VDI. Virtual device interface.

virtual disk(ette). A portion of a disk attached to the host system that is used to emulate a IBM Personal System/2 disk(ette). A virtual diskette has a capacity of up to 320 K. A virtual disk has a capacity of up to 32 MB.

withdraw. (1) To remove. (2) To make no longer available.

work station. An input/output device consisting of a display station and/or printer that allows

you to send and receive information from a host system.

work station address. The address assigned to a work station enabling the host system to communicate with it.

work station line. The twinaxial cable from a host computer or the chain of work stations to which a work station is connected.

WP. Word Processing.

Index

A

address

See next session address
number

See station address function

application program

example 1-3
interface access 1-4
interface with Emulation 1-3
security viii
subroutine entry point 1-5
warning viii

application screen save and restore

buffer pointer 1-17

arctic device

model function 1-24
type function 1-24

ASCII

character buffer function 1-7
display table 1-12
printer table 1-22
transparent data command
5219 3-5
5224 3-13
update display 1-6

ASCII character set

codepage 437 A-21
codepage 850 A-22
codepage 860 A-23
codepage 863 A-24

ATTDEVID 1-24

attribute appearances

color displays 2-6
monochrome displays 2-2

attribute table pointer 1-13

automatic Hot Key

bypass function 1-16
function 1-20

B

BASIC

program interface access 1-4
sample program 1-4
subroutine restriction 1-5

baud rate 1-27

blink attribute 6-5

buffer

byte count for 1 1-25
byte count for 2 1-26
character function 1-7
EBCDIC screen 1-13
flag byte pointer 1-25
pointer to restore 1-17
pointer to 1 1-25
pointer to 2 1-25
regeneration 1-2

BUFxFULL@ 1-25

bypass presign-on screen function 1-16

C

character sets 3-2, 4-13

ASCII tables A-21
EBCDIC tables A-11
EBCDIC-to-ASCII
conversion A-25
printer character
conversion A-2
scan codes
IBM Personal System/2 A-3
IBM 5250 A-7

characters per inch 3-3

characters per line 3-3

CODESEG 1-21, 1-24

color displays 2-6

color mapping function 6-3

- column separator attribute 6-5
- COM ports 1-26
- comment command (update) C-3
- compressed print 3-3
- continuous forms 3-3
- control data set for updating C-2
- control of graphics displays
 - function 6-3
- control sequences, entering in a
 - PFT 4-4
- CPI
 - See characters per inch
- CPL
 - See characters per line
- CRT-SAV 1-17
- CURSOR 1-17
- cursor position function 1-17

D

- data follows order 5-30
- data security viii
- data segment value function 1-6
- decimal to hexadecimal
 - conversion B-1
- default type style 4-13
- DEFAULT.PFT 4-2
- device driver specifications 6-6
- device type
 - display model ID byte 1-18
 - display type ID byte 1-18
 - host printer 1-24
 - identifier 1-6
 - printer 1-24
 - printer model 1-24
- displays
 - control 1-2
 - differences
 - color 2-6
 - monochrome 2-2
 - graphics 6-3
 - screen function 1-9
 - session number 1-21
 - type pointer 1-19

- drive intercept 1-14
- DSKDI 1-14
- DSKDI0 1-14
- DSKDI1 1-15
- DSKDI2 1-15
- DVC-TYP 1-18, 1-24

E

EBCDIC character set

- codepage 037 A-11
- codepage 273 A-12
- codepage 277 A-13
- codepage 278 A-14
- codepage 280 A-15
- codepage 282 A-16
- codepage 284 A-17
- codepage 285 A-18
- codepage 297 A-19
- codepage 500 A-20
- viewing A-25

EBCDIC screen buffer pointer 1-13

EBCDIC-to-ASCII table pointer

- display 1-12
- printer 1-22

Emulation program

- accessing from a BASIC
 - program 1-4
- description v
- display control 1-9
- fixed address vectors to key
 - emulator functions 1-4
- functions
 - display 1-11
 - general purpose 1-5
 - printer 1-22
- interface
 - application program 1-3
 - command 1-2
 - host 1-2
 - operator 1-2
- modifying 1-3
- operating from application
 - program 1-3

Emulation program (continued)

release level 1-11
 segment address
 display 1-21
 printer 1-24
 subroutine entry point 1-5
 updating C-1

End of Document Reset 4-8
engineering change level 1-11,
1-14, 1-22

Enhanced keyboard A-3
error conditions 3-15

F**fixed address vector table, display**

CODESEG 1-21
 CRT-SAV 1-17
 CURSOR 1-17
 DSKDI 1-14
 DSKDI0 1-14
 DSKDI1 1-15
 DSKDI2 1-15
 DVC-TYP 1-18
 FLAGS 1-20
 HKCHAR 1-20
 HKCOL 1-20
 HKROW 1-20
 INDIC 1-15
 INDICRS 1-12
 INDICST 1-12
 KEYI 1-14
 KEYO 1-14
 LASTKEY 1-18
 LKTIMSK 1-14
 MODEL-ID 1-18
 NOTMSK 1-14
 NSDS 1-21
 PCREGN@ 1-19
 PRTBC 1-18
 SESSNO 1-21
 STN@ 1-21
 SURX 1-18
 SYSAV 1-15

**fixed address vector table, display
(continued)**

VRDO 1-18
 XEC 1-14
 XILVL@ 1-14
 XREGEN 1-13
 XXLATT 1-13
 XXLCHR 1-12
 XXLKEY 1-13

fixed address vector table, printer

ATTDEVID 1-24
 BUFXFULL@ 1-25
 CODESEG 1-24
 DVC-TYP 1-24
 INDICAT@ 1-28
 INITSTRNG@ 1-23
 MESSAG@ 1-29
 MODEL-ID 1-24
 NSDS 1-25
 PTRDEVNAM 1-26
 RCBUF1@ 1-25
 RCBUF1CT@ 1-25
 RCBUF2@ 1-25
 RCBUF2CT@ 1-26
 SERPARM 1-27
 SESSNO 1-24
 STN@ 1-24
 SURX 1-23
 XEC 1-22
 XXLATT 1-22
 XXLCHR 1-22

FLAGS 1-20**forms handling 3-3****function selection tests 4-15****G****general purpose subroutine 1-5**

entry point 1-5
 ILHOTKEY 1-9
 ILNOSHOW 1-9
 ILPRTCHAR 1-7
 ILPRTOPTN 1-8
 ILQRESET 1-7

general purpose subroutine 1-5

(continued)

ILSAVE 1-10

ILSCRNUP 1-6

ILSELSSEN 1-6

go to standby order 5-32

graphics

device driver specifications 6-6

display differences 6-3

blink, underscore and

column separators 6-5

color mapping 6-3

control of graphics

displays 6-3

graphics coordinates 6-3

line style 6-3

line style mapping 6-3

local select modes 6-5

polygon fill 6-4

scanline data 6-4

set line weight order 6-4

status line control 6-4

styled line 6-4

text attributes and

colors 6-4

vertical scaling factor 6-5

write background 6-5

write polymarker 6-5

XOR and OR writing

modes 6-5

graphics control orders 5-5

graphics coordinates function 6-3

graphics display off order 5-7

graphics display on order 5-6

graphics draw orders 5-21

graphics orders

fully supported and no-op

orders 5-4

graphics control orders 5-5

graphics display off 5-7

graphics display on 5-6

graphics draw orders 5-21

write background 5-22

graphics set orders 5-8

set attributes 5-20

graphics orders (continued)

graphics set orders 5-8

(continued)

set color 5-11

set color table 5-8

set fill mode 5-17

set function 5-15

set style 5-13

IEEE interface orders 5-29

data follows 5-30

go to standby 5-32

interface clear 5-32

take control

asynchronously 5-31

level of support 5-2

printer orders 5-28

read orders 5-23

read attributes 5-26

read status 5-23

graphics set orders 5-8

group type styles 4-14

H

hexadecimal to decimal

conversion B-1

highlighting, printer 4-11

HKCHAR 1-20

HKCOL 1-20

HKROW 1-20

horizontal character spacing 4-9

horizontal relative movement 4-10

host printer commands

forms handling 3-3

5219 3-5

5224 3-12

host system interface

connection 1-2

data transmission 1-2

Hot Key

automatic 1-16

automatic, function 1-20

character 1-20

column 1-20

Hot Key (continued)

control flags 1-20
control function 1-23
function 1-9
row 1-20

I

IBM Personal System/2 keyboard and scan codes

See character sets, scan codes

IBM3812.PFT 4-2
IBM4202.PFT 4-2
IEEE data follows order 5-30
IEEE interface orders 5-29
ILHOTKEY 1-9
ILNOSHOW 1-9
ILPRTCHAR 1-7
ILPRTOPTN 1-8
ILQRESET 1-7
ILSAVE 1-10
ILSCRNUP 1-6
ILSELSSEN 1-6
indexing functions 4-9
INDIC 1-15
INDICAT@ 1-28
indicator state flag 1-28
indicator status byte 1-15
INDICRS 1-12
INDICST 1-12
individual type style 4-13
INITSTRNG@ 1-23
Input Inhibited indicator 1-12
Insert mode, cancelling 1-7
interface clear order 5-32
interrupt mask byte 1-14
 inversion 1-14
interrupt vector address
 pointer 1-14

J

justification 4-10

K

keyboard buffer clear function 1-7

KEYI 1-14

KEYO 1-13, 1-14

L

LASTKEY 1-18

line style 5-13

 function 6-3

 mapping 6-3

 order 5-14

lines per inch 3-3, 4-8

LKTIMSK 1-14

Local Select mode 6-5

LPI

 See lines per inch

LPT ports 1-26

M

MESSAG@ 1-29

message state flag 1-29

MODEL-ID 1-18, 1-24

monochrome displays 2-2

Multi-language Enhanced
 keyboard A-4

N

next session address number

 display 1-21

 printer 1-25

NOTMSK 1-14

NSDS 1-21, 1-25

P

page length 4-12

page presentation mode
 command 3-9

paper feed 4-11

paper handling 4-11

paper length 3-3

paper width

- continuous form 3-3
- single sheet 3-3

parity 1-27**PC scan codes**

- See character sets, scan codes

PCREGN@ 1-19**PFTs, provided by the setup****program 4-2****PFTWSN.PRN 4-2****playback sequence 1-7****polygon fill function 6-4****port type 1-26****presign-on screen bypass****function 1-16****print line length 3-3****print screen character set 1-18****printer attribute table pointer 1-22****printer character conversion A-2****printer character set boundary****check value 1-18****printer control codes buffer****function 1-7****printer control sequences in a****PFT 4-4****printer device name 1-26****printer device type function 1-24****printer error conditions 3-15****printer function table****creating a PFT 4-3****defining printer functions 4-6****default type style 4-13****End of Document Reset 4-8****group type styles 4-14****highlighting 4-11****horizontal character****spacing 4-9****horizontal relative****movement 4-10****indexing functions 4-9****individual type style 4-13****lines per inch 4-8****page length 4-12****paper handling 4-11****proportional space mode****and justification 4-10****printer function table (continued)****defining printer functions 4-6**
(continued)**Start of Document****Initialization 4-7****type styles and character**
sets 4-13**variable line spacing 4-9****vertical and horizontal****spacing 4-8****vertical line spacing 4-8****entering control sequences 4-4****variable n 4-5****function selection tests 4-15****PFTs provided by the****program 4-2****DEFAULT.PFT 4-2****IBM3812.PFT 4-2****IBM4202.PFT 4-2****worksheet 4-2****printer functions, defining in a****PFT 4-6****printer initialization string****pointer 1-23****printer operator panel****indicators 1-28****messages 1-29****options 1-8****refresh 1-23****printer orders 5-28****printer session number 1-24****printers, using****character sets 3-2****characteristics of host****printers 3-3****EBCDIC-to-ASCII****translation 3-4****forms handling 3-3****5219 commands 3-5****ASCII transparent data 3-5****page presentation****mode 3-9****set CGCS through local****ID 3-11****set character density 3-8**

printers, using (continued)

5219 commands 3-5

(continued)

set FID through GFID 3-7

set GCGID through

GCID 3-11

set justify mode 3-7

set line density 3-6

set line spacing 3-7

set presentation page

size 3-9

set printer setup 3-11

set single line distance 3-8

set vertical format 3-6

set vertical margins 3-10

subscript 3-11

superscript 3-5

unit backspace 3-6

5224 commands 3-12

ASCII transparent data 3-13

commands not

supported 3-12

set character density 3-15

set line density 3-14

set vertical format 3-14

printwheel 3-2, 4-13

proportional space mode 4-10

PRTBC 1-18

PTRDEVNAM 1-26

R

RCBUF1@ 1-25

RCBUF1CT@ 1-25

RCBUF2@ 1-25

RCBUF2CT@ 1-26

read attributes order 5-26

read orders 5-23

read status order 5-23

refresh screen function 1-6

registers

AH 1-5

AL

See general purpose
subroutine

registers (continued)

CX

See general purpose
subroutine

DS

See general purpose
subroutine

ES:DI

See general purpose
subroutine

saving 1-5

related manuals vii

release level 1-11

release level pointer 1-14, 1-22

reset command (update) C-3

restore buffer pointer 1-17

S

save screen function 1-10

scan code 1-14, A-3

description 1-2

IBM Personal System/2 1-14

IBM Personal System/2

table A-5

queue clear function 1-7

table pointer 1-13

5250 1-14

5250 table A-8

5250, last 1-18

scanline data function 6-4

screen change flag function

display 1-18

printer 1-23

screen refresh function 1-6

screen save function 1-10

screen save pointer 1-17

security viii

segment address of Emulation

program code

display 1-21

printer 1-24

serial printer parameter byte
 baud rate 1-27
 parity 1-27
 stop bit 1-27
 word length 1-27
SERPARM 1-27
session number function 1-21, 1-24
session, next, address number
 display 1-21
 printer 1-25
SESSNO 1-21, 1-24
set attributes order 5-20
set CGCS through local ID
 command, 5219 3-11
set character density command
 5219 3-8
 5224 3-15
set color order 5-11
set color table order 5-8
set FID through GFID command,
 5219 3-7
set fill mode order 5-17
set function order 5-15
set GCGID through GCID command,
 5219 3-11
set justify mode command,
 5219 3-7
set line density command
 5219 3-6
 5224 3-14
set line spacing command,
 5219 3-7
set line weight order function 6-4
set presentation page size
 command, 5219 3-9
set printer setup command,
 5219 3-11
set single line distance command,
 5219 3-8
set style order 5-13
set vertical format command
 5219 3-6
 5224 3-14

set vertical margins command,
 5219 3-10
Shift indicator 1-12
single sheet forms 3-3
stamp command (update) C-3
Start of Document Initialization 4-7
station address function
 display 1-21
 printer 1-24
status indicator
 display 1-15
 display function 1-12
 display history function 1-12
 printer 1-28
status line control function 6-4
STN@ 1-21, 1-24
stop bit 1-27
styled line function 6-4
subroutine
 See also general purpose
 subroutine
 sample program to access 1-11
subscript command, 5219 3-11
superscript command, 5219 3-5
SURX 1-18, 1-23
SYSAV 1-15
System Available indicator 1-15

T

take control asynchronously
 order 5-31
technical considerations
 application program
 interface 1-3
 BASIC application program
 interface access 1-4
 sample BASIC program 1-4
 fixed address vector table
 See also fixed address
 vector table, display
 See also fixed address
 vector table, printer
 display 1-11
 printer 1-22

technical considerations (continued)

- fixed address vectors to key emulator functions 1-4
- for printer sessions 1-22
- general purpose subroutine 1-5

See also general purpose subroutine

- interface with host 1-2
- sample program 1-11

text attributes and colors function 6-4

transparent data command

- 5219 3-5
- 5224 3-13

type styles

- default 4-13
- group 4-14
- individual 4-13

U

U.S.A. Enhanced keyboard A-3

underscore attribute 6-5

unit backspace command, 5219 3-6

update command C-3

updating the emulation program

- control data set C-2
 - valid commands C-3
- example of a control data set C-4
- listing applied updates C-6
- updating with the Update program C-5
 - listing applied updates C-6

V

variable line spacing 4-9

- variable n 4-5
 - format 4-6

VDI device drivers 6-6

verify command (update) C-4

vertical and horizontal spacing 4-8

vertical line spacing 4-8

vertical scaling factor function 6-5

virtual diskette drive

- drive 1 intercept 1-14
- drive 2 intercept 1-14
- drive 3 intercept 1-15
- intercept override 1-18
- name function 1-14

virtual versus real diskette intercept request override 1-18

VRDO 1-18

W

word length 1-27

worksheet for PFT setup program 4-2

write background function 6-5

write background order 5-22

write polymarker function 6-5

writing mode 5-15

WSEASO.BAS 1-14

See also Chapter 2 in the User's Guide

WSESS.BAS 1-13

See also Chapter 2 in the User's Guide

WSESVD.BAS 1-14

See also Chapter 2 in the User's Guide

WSEUL.BAS 1-13

See also Chapter 2 in the User's Guide

X

XEC 1-14, 1-22

XILVL@ 1-14

XOR and OR writing modes function 6-5

XREGEN 1-13

XXLATT 1-13, 1-22

XXLCHR 1-12, 1-22
XXLKEY 1-13

Numerics

037 codepage A-11
273 codepage A-12
277 codepage A-13
278 codepage A-14
280 codepage A-15
282 codepage A-16
284 codepage A-17
285 codepage A-18
297 codepage A-19
3812 Pageprinter 4-2
4202 Proprinter XL 4-2
437 codepage A-21
500 codepage A-20

5216 printwheel 3-2

5219

bold 4-11

printer commands 3-5

See also printers, using

unit backspacing 4-10

5224

Advanced Printer Function

program 3-12

commands not supported 3-12

printer commands 3-12

See also printers, using

5250 keyboard and scan codes

See character sets, scan codes

5250 scan codes A-8

850 codepage A-22

860 codepage A-23

863 codepage A-24

READER'S COMMENT FORM

Please use this form only to identify publication errors or to request changes in publications. Direct any requests for additional publications, technical questions about IBM systems, changes in IBM programming support, and so on, to your IBM representative or to your nearest IBM branch office. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

☐

If your comment does not need a reply (for example, pointing out a typing error) check this box and do not include your name and address below. If your comment is applicable, we will include it in the next revision of the manual.

☐

If you would like a reply, check this box. Be sure to print your name and address below.

Please contact your nearest IBM branch office to request additional publications.

Name

Company or
Organization

Address

City State Zip Code

Phone Number
(Area Code)

Page Number(s):

Comment(s):



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS / PERMIT NO. 40 / ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corp.
Information Development
Department 245
Rochester, Minnesota, U.S.A. 55901



Fold and tape. **Please do not staple.**

IBM

READER'S COMMENT FORM

Please use this form only to identify publication errors or to request changes in publications. Direct any requests for additional publications, technical questions about IBM systems, changes in IBM programming support, and so on, to your IBM representative or to your nearest IBM branch office. You may use this form to communicate your comments about this publication, its organization, or subject matter, with the understanding that IBM may use or distribute whatever information you supply in any way it believes appropriate without incurring any obligation to you.

- ☐ If your comment does not need a reply (for example, pointing out a typing error) check this box and do not include your name and address below. If your comment is applicable, we will include it in the next revision of the manual.
- ☐ If you would like a reply, check this box. Be sure to print your name and address below.

Please contact your nearest IBM branch office to request additional publications.

Name

Company or
Organization

Address

City State Zip Code

Phone Number
(Area Code)

Page Number(s):

Comment(s):



NO POSTAGE
NECESSARY
IF MAILED IN THE
UNITED STATES

BUSINESS REPLY MAIL

FIRST CLASS / PERMIT NO. 40 / ARMONK, NEW YORK

POSTAGE WILL BE PAID BY ADDRESSEE

International Business Machines Corp.
Information Development
Department 245
Rochester, Minnesota, U.S.A. 55901



Fold and tape. **Please do not staple.**

IBM





International Business Machines Corporation

Hwy 52 N.W. 37th Street
Rochester, Minnesota 55901

Order Number
SC21-8068-0

Part Number
72X6152